CHAPTER 8

NETWORK OPERATING SYSTEMS AND WINDOWS 2000-BASED NETWORKING

After reading this chapter and completing the exercises, you will be able to:

- Discuss the functions and features of a network operating system
- Define the requirements for a Windows 2000 network environment
- Describe how a Windows 2000 server fits into an enterprise-wide network
- Perform a simple Windows 2000 Server installation
- Manage simple user, group, and rights parameters in Windows 2000 Server
- Understand how Windows 2000 Server integrates with other popular network operating systems

ON THE JOB

A few years ago I started working for a large consumer goods distributor that relied on Windows NT Server to run its truck distribution software. This software performed route mapping, allowed drivers to scan their bills of lading and receipts, and also kept our company's inventory. More than 100 users used it heavily, day and night. We found that Windows NT was sometimes unstable. We added extra servers to take over in case one of the servers froze up during the night, but we still had problems to resolve in the morning.

Because of our problems, we were eager to try Windows 2000 Server when it was released. We waited until it had been out a few months and the magazine reviews looked positive. We tried it on a test server. Right away we noticed how much more reliable this operating system was compared to Windows NT. We decided to upgrade all of our servers to Windows 2000, and we haven't looked back.

Bob Steigel
Mercury Foods

Network operating systems enable servers to share resources with clients. They also facilitate other services such as communications, security, and user management. Network operating systems do not fit neatly into one layer of the OSI Model. Some of their functions—those that facilitate communication between computers on a network—belong in the Application layer. However, many of their functions—those that interact with users—take place above the Application layer (that is, above the top layer) of the OSI Model. Consequently, the OSI Model does not usefully describe all aspects of network operating systems.

During your career as a networking professional, you will probably work with more than one network operating system (NOS). At the same time, you may work with several versions of the same network operating system. To qualify for Net+ certification, you must understand the inner workings of network operating systems in general. In addition, you must be familiar with the three major network operating systems (Windows 2000 Server, NetWare, and UNIX) and be able to discuss their similarities and differences. Finally, you must be able to integrate the major operating systems, when necessary.

This chapter introduces the basic concepts related to network operating systems and discusses in detail one of the most popular network operating systems, Windows 2000 Server. The following two chapters focus on NetWare and UNIX.

Introduction to Network Operating Systems

So far you have focused on the foundations of networking—that is, the lower layers of the OSI Model. Specifically, you've learned about the Physical layer up through the middle layers, where protocol addressing, error checking, and session negotiation occur. Now you will tackle the upper layers of the OSI Model, without which you could not control the lower layers.

Some pieces of network operating systems belong to the seventh layer of the OSI Model, the Application layer. This layer allows you to control the transmission of data by taking your requests and translating them into instructions to the lower-layer processes. The pieces of a network operating system that provide the user interface (such as the dialog boxes in Windows 2000 Server with which you can add users or create groups of users) actually belong *above* the Application layer. Strictly speaking, then, a network operating system does not belong entirely to the Application layer, but rather straddles the Application layer and an imaginary eighth layer (which the OSI Model does not address) above the Application layer.

Recall from Chapter 1 that most modern networks are based on a client/server architecture, in which a server enables multiple clients to share resources. Once installed on a server, a network operating system can oversee user and group management, central data storage, file and print sharing, communications, security, and messaging. It may also support many other functions, such as Internet and remote connectivity, network management, and data backup and recovery. Network operating systems are entirely software-based and can run on a number of different hardware platforms and network topologies.

When installing a network operating system, you may accept the default settings or customize your configuration to more closely meet your needs. You may also take advantage of special services or enhancements that come with a basic network operating system. For example, if you install Windows 2000 Server with only its minimum components, you may later choose to install a clustering solution so that multiple servers can share the network's processing burden. The components included in each different network operating system

and every version of a particular network operating system vary. This variability is just one reason that you should plan your network operating system installation very carefully before beginning the implementation. The myriad ways to install and configure network operating systems are beyond the scope of this book.

In this chapter, the word "server" refers to the hardware on which a network operating system runs. In the field of networking, the word "server" may also refer to an application that runs on this hardware to provide a dedicated service. For example, although you may use a Compaq server as your hardware, you may run Novell's BorderManager application as your proxy server on that hardware.

Although each network operating system discussed in this book supports file and print sharing, plus a host of other services, network operating systems differ in how they achieve those functions, what type of environment they suit, and how they are administered. In the next section, you will learn how to select a network operating system for your network.

Selecting a Network Operating System

Realistically, when designing a local area network, you can select from only a handful of network operating systems—specifically, Windows NT Server, Windows 2000 Server, NetWare, and some version of UNIX or Linux. The only reason not to choose one of these options is if your network is outdated or runs a proprietary, specialized application (for example, a quality control system that measures performance of catalytic converters in a test laboratory) that requires a less familiar network operating system (such as Banyan VINES). Many LAN environments include a mix of all three major network operating systems, making interoperability a significant concern.

When choosing a network operating system, you should certainly weigh the strengths and weaknesses of the available options before making a choice. Nevertheless, your decision will probably depend largely on the operating systems and applications already running on the LAN. In other words, your choice may be limited by the existing infrastructure. (Infrastructure includes other network operating systems, and also LAN topology, protocols, transmission methods, and connectivity hardware.)

For example, suppose that you are the network manager for a community college that uses 150 NetWare 4.11 servers to manage all IDs, security, and file and print sharing for 4000 users. In addition, you oversee five Windows 2000 servers that provide Web development and backup services. You have been asked to select a network operating system for a new server for the college's Theater Department. You probably wouldn't choose Windows 2000 Server, because a NetWare server would integrate more seamlessly with your existing network and facilitate administrative tasks, such as adding new users or resources. At another organization, the opposite situation may prevail.

The following list summarizes the questions you should ask when deciding to invest in a network operating system. You need to weigh the importance of each factor in your organization's environment separately.

- Is it compatible with my existing infrastructure?
- Will it provide the security required by my resources?
- Can my technical staff manage it effectively?
- Will my applications run smoothly on it?
- Will it accommodate future growth (that is, is it scalable)?
- Does it support the additional services my users require (for example, remote access, Web site hosting, and messaging)?
- Does it fit my budget?
- Can I count on competent and consistent support from its manufacturer?

The importance of these concerns will vary from one network administrator to the next. For example, imagine that you are the network administrator for a multinational chemical company with locations across the globe. Your company's plants and profitability may depend on your network always being available, and your IT budget may be large. In this case, the cost of a network operating system may be less important than its ability to accommodate future growth and the availability of the vendor's technical support. In contrast, if you were the network administrator for a local nonprofit food pantry, your greatest concern may be the cost of the network operating system. In this case, you probably won't care whether the system can easily grow to support hundreds of servers.

In addition to assessing each NOS according to your needs, you should test your network operating system choice in your environment before making a purchase. You can perform such testing on an extra server, using a test group of typical users and applications with specific test criteria in mind. Chapter 16 discusses the implementation of test (or "pilot") networking systems. Bear in mind that you cannot rely on trade magazine articles or a vendor's marketing information to accurately predict which network operating system will best suit your circumstances.

Network Operating Systems and Servers

Most networks include servers that exceed the minimum hardware requirements suggested by the software vendor. Every situation will vary, but to determine the optimal hardware for your servers, you should consider the following issues:

- How many clients will connect to the server?
- What kinds of applications will run on the server?
- How much storage space will each user need?

- How much downtime is acceptable?
- What can the organization afford?

Perhaps the most important question in this list involves the types of applications to be run by the server. You can purchase an inexpensive, low-end server that runs Windows 2000 Server adequately, but that will suffice only for file and print sharing. To perform more functions with your network, you would need to invest in a server that has sufficient processing and memory resources to run applications as well. The particular type of server you choose will depend on the applications you want to run. As you can imagine, every application comes with different processor, RAM, and storage requirements. (Consult the application's installation guide for specifics.) In general, you can assume that a database program (such as MS SQL Server) will require more processor and RAM resources than a word-processing application (such as MS Word).

Keep in mind that the particular way an application uses resources may influence your choice of software and hardware. Applications may or may not provide the option of sharing the processing burden between the client and server. For example, you might install a group scheduling and messaging package that requires every client to run executable files from a network drive, thereby almost exclusively using the server's processing resources. Alternately, you may install the program files on each client workstation and use the server only to distribute messages. The latter solution puts the processing burden on the client.

If your server assumes most of the application-processing burden, or if you have a large number of services and clients to support, you will need to add more hardware than the minimum network operating system requirements. For example, you might add multiple processors, as discussed in the next section. You might also install more RAM, multiple NICs, fault-tolerant hard disks, a backup drive, and an uninterruptible power supply (UPS). Each of these components will enhance network reliability or performance. (You will learn more about performance and reliability in Chapters 13 and 14.) For now, it suffices to know that you must carefully analyze your current situation and plans for growth before making a hardware purchasing decision. Whereas high-end servers with massive processing and storage resources plus fault-tolerant components can cost as much as $100,000, your department may need only a $2000 server. No matter what your needs, you should ensure that your hardware vendor has a reputation for high quality, dependability, and excellent technical support. Although you may be able to trim your costs on workstation hardware by using generic models, you should spare no expense in purchasing your server. A component failure in a server can cause problems for many people, whereas a workstation problem will probably affect only one person.

Network Operating System Services and Features

By now you are familiar with the basic functions that network operating systems provide, including resource sharing, security, and network management. In this section you will learn more about fundamental NOS functions and the meaning of terms used when comparing NOSs. You will also learn about some advanced features that enable NOSs to service clients more quickly and reliably. These features are available in all of the popular NOSs. However, the degree to which each NOS can support these features may differ. As you read about Windows 2000 Server in this chapter, and NetWare and UNIX in later chapters, you will learn more about their differences.

Client Support

The primary reason for using networks is to enable clients to communicate and share resources efficiently. Therefore, client support is one of the most important functions provided by an NOS. For purposes of this discussion, client support includes the following tasks:

- Creating client accounts and enabling them to connect to the network
- Managing client accounts
- Enabling clients to share resources
- Managing client access to shared resources
- Enabling clients to communicate with other clients

In Chapter 3 you learned how clients and servers communicate through the layers of the OSI Model (recall the example of sending an e-mail message). The following discussion provides a more general view of client/server communication.

Client/server Communication

Both the client software and the network operating system participate in logging a client onto the server. Although client software differs according to the NOS and desktop operating system, the process of logging on is similar no matter what software is used. First, the user launches the client software from his desktop. Then he enters his user name and password and presses the Enter key. At this point a service on the client workstation (called the **redirector**) intercepts the request to determine whether it should be handled by the client or by the server. A redirector, which belongs to the Presentation layer of the OSI Model, is a service of both the network operating system and the client operating system. Once the client's redirector decides that the request is meant for the server, the client transmits this data over the network to the server. (If the redirector had determined that the request was meant for the client, rather than the server, it would have issued the request to the client's CPU.) For security's sake, most modern clients will encrypt user name and password information before transmitting it to the network media. Recall from Chapter 2 that encryption is another Presentation layer function.

At the server, the network operating system receives the client's request for service and unencrypts it, if necessary. It attempts to match the user name to a name in its user database. If it is successful, it then compares the password associated with that user name to the password supplied by the user. If the passwords match, the NOS responds to the client by granting it access to resources on the network, according to limitations it has specified for this client. This process is known as **authentication**. Figure 8-1 depicts the process of a client connecting to a network operating system.

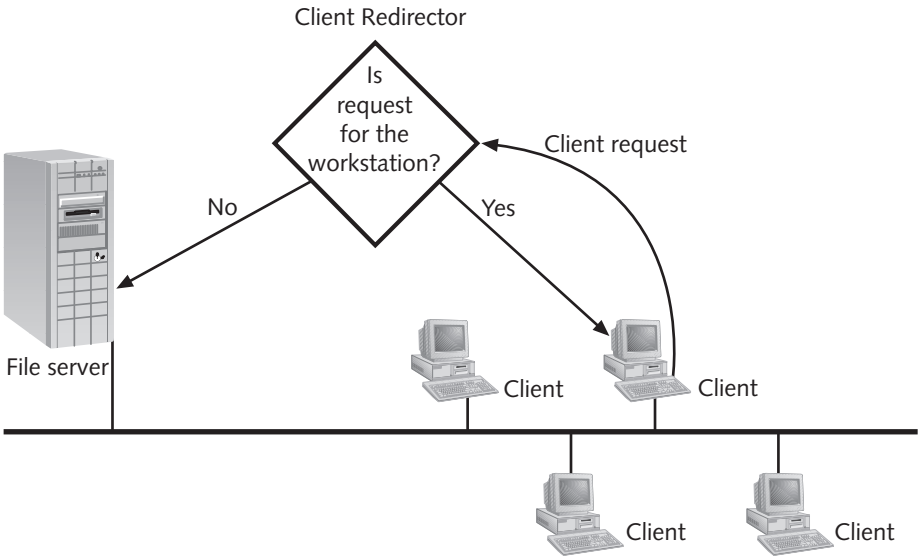

Figure 8-1 A client connecting to a network operating system

You should understand the login process for troubleshooting purposes. For example, if after entering her name and password, a user receives an error message indicating that the server was not found, you can conclude that the request never made it to the server's NOS. In this case a physical connection problem may be at fault. However, if after entering her name and password, a user receives an error message indicating that the user name or password is invalid, you know that at least the physical connection is working because the request reached the NOS and the NOS attempted to verify the user name. In this case, the password or user name may have been typed incorrectly.

Once the client has successfully logged on, the client software communicates with the network operating system each time the client requests services from the server. For example, if you wished to open a file on the server's hard disk, you would interact with your workstation's operating system to make the file request; the file request would then be intercepted by the redirector and passed to the server via the client software.

In some instances a piece of software called **middleware** is necessary to translate requests and responses between the client and server. Middleware may be used as a messaging service between clients and servers, as a universal query language for databases, or as means of coordinating processes between multiple servers that need to work together in servicing clients. For example, a library's database of materials is contained on a UNIX server, and multiple workstations using different client OSs need to access that single database. In this case, middleware can enable the multiple types of clients to access the database in one standard format. This prevents the library from having to install multiple instances of their database, one for each different client platform. It also allows the clients to have little responsibility for processing requests and therefore, use little of the client's resources. A client/server environment that uses middleware in this fashion is also known as a **3-tier architecture**, because of its three layers: client, middleware, and server. To take advantage of a 3-tier architecture a client workstation requires a special type of software known as a **thin client**. Middleware typically runs on the server, but may run on both the server and the client. Figure 8-2 illustrates the concept of middleware.

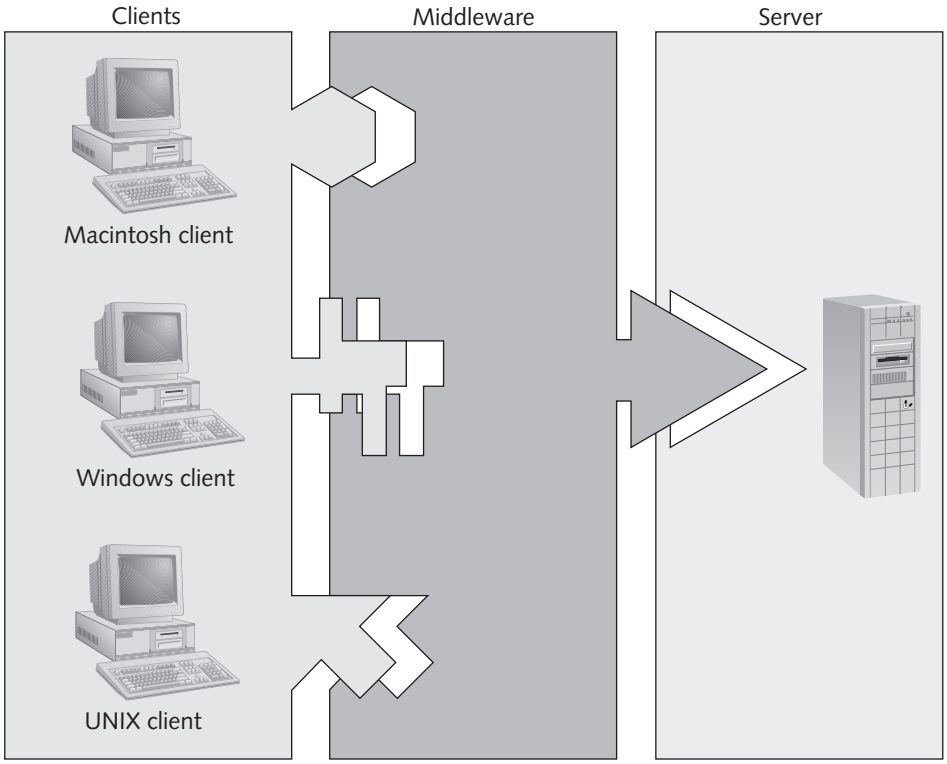

Figure 8-2 Middleware between clients and a server

In the early days of networking, client software from one manufacturer could not always communicate with network software from another manufacturer. Now, however, most every type of client can log onto most every type of network operating system. In some cases this communication may require a special utility or third-party software. But usually, the NOS manufacturer supplies a preferred client software package for each popular type of client. For example, Novell recommends installing its "Novell Client for Windows 95/98" on Windows 95 or Windows 98 workstations and its "Novell Client for Windows 2000" on Windows 2000 workstations. Microsoft requires the "Client for Microsoft Networks" for Windows workstations connecting to its Windows 2000 Server NOS. Client software other than that recommended by the NOS manufacturer may work, but it is wise to follow the NOS manufacturer's guidelines.

Users and Groups

After a client is authenticated by the NOS, it is granted access to services and resources managed by the NOS. The type of access a client (or user) has depends on her user account and what groups she belongs in. In this section, you will learn about users and groups of users. Later in this chapter (and in Chapters 9 and 10), you will learn how to create users and groups and give them rights to resources in each of the three common NOSs. You will also learn which tools each NOS provides for managing users and groups.

You have probably worked with enough computers and networks to know why user names are necessary: to grant each user on a network access to files and other shared resources. Imagine that you are the network administrator for a large college campus with 20,000 user names. Assigning directory, file, printer, and other resource rights for each user name would consume all of your time, especially if the user population changed regularly. To more easily manage network access, you can combine users with similar needs and restrictions into **groups**.

Groups form the basis for resource and account management for every type of network operating system. Many network administrators create groups according to department or, even more specifically, according to job function within a department. They then assign different file or directory access rights to each group. For example, on a high school's network, the administrator may create a group called "students" for the students and a group called "teachers" for teachers. The administrator could then easily grant the teachers group rights to view all attendance and grade records on the server, but deny the same access to the students group.

To better understand the role of groups in resource sharing, first consider their use on a relatively small scale. Suppose you are the network administrator for a public elementary school. You might want to give all teachers and students access to run instructional programs from a network directory called PROGRAMS. In addition, you might want to allow teachers to install their own instructional programs in this same directory. Meanwhile, you need to allow teachers and administrators to record grade information in a central database called GRADES. Of course, you don't want to allow students to

read information from this database. Finally, you might want administrators to use a shared drive called STAFF to store performance review information, which should not be accessible to instructors or students. Table 8-1 illustrates how you can provide this security by dividing separate users into three groups: instructors, students, and administrators.

Table 8-1 Providing security through groups

Group	Rights to PROGRAMS	Rights to GRADES	Rights to STAFF
Instructors	Read, modify	Full control	No access
Students	Read	No access	No access
Administrators	No access	Read, modify	Full control

Plan your groups carefully. Creating many groups (for example, a separate group for every job classification in your organization) may impose as much of an administrative burden as not using any groups.

As you learned earlier, once an NOS authenticates a user, it checks the user name against a list of resources and their access restrictions list. If the user name is part of a group with specific access permissions or restrictions, the system will apply those same permissions and restrictions to the user's account.

For simpler management, groups can be nested (one within another) or arranged hierarchically (multiple levels of nested groups) according to the type of access required by different types of users. The way groups are arranged will affect the permissions granted to each group's members. For example, if you created a group called Temps within the Administrators group for temporary office assistants, the Temps group would be nested within the Administrators group and would, by default, share the same permissions as the Administrators group. If you wanted to restrict the Temps users from seeing the staff performance reviews, you would have to separately assign restrictions to the Temps group for that purpose. Once you assign different rights to the Temps group, you have begun creating a hierarchical structure of groups. NOSs differ slightly in how they treat nested and hierarchical groups, and enumerating these differences is beyond the scope of this book. However, if you are a network administrator, you must thoroughly understand the implications of complex group arrangements. For the Network+ exam, you should at least understand how groups can be used to efficiently manage permissions and restrict or allow access to resources.

Once the user and group restrictions are applied, the client is allowed to share resources on the network, including data, data storage space, applications, and peripherals. You will learn more about resource sharing later in this chapter. To understand how NOSs enable resource sharing, though, it is useful to first understand their fundamental design, beginning with their directories.

Directories

A **directory** is a list that organizes resources and associates them with their properties, or characteristics. For example, the table of contents of this book is an example of a directory. In this analogy, each chapter is a resource; the page numbers associated with each heading in a chapter are properties of the chapter. To determine where the book discusses network operating systems, you would find "network operating systems" in the table of contents, then note what pages were listed for this topic. On a personal computer, a directory provides information about the way in which files are organized, plus information about those files, such as their size and creation date.

In the context of the Windows 2000 Server and NetWare 5.x network operating systems, a directory is a method for organizing and managing objects. An **object** is a representation of a thing or person associated with the network. Objects commonly managed by NOS directories include users, printers, groups, computers, data files, and applications. Each object may have a multitude of **attributes**, or properties, associated with it. For example, a user's attributes may include a first and last name, location, mail address, group membership, access restrictions, and so on. A printer's attributes may include an administrator, location, model number, printing preferences (for example, double-sided printing), and so on.

To better organize and manage objects, a network administrator places objects in containers. **Containers** are logically defined receptacles that serve only to assemble similar objects. Returning to the example of a school network, suppose each student, teacher, and administrator were assigned a user name and password for the network. Each of these users would be considered an object, and each would require an account. (An **account** is the record of a user that contains all of his or her properties, including rights to resources, password, name, and so on.) One way of organizing these objects would be to put all the user objects in one container called "Users." But suppose the school provided a server and a room of workstations strictly for student use. The use of these computers would be restricted to applications and Internet access during only certain hours of the day. As the network administrator, you could gather the student user names (or the "Students" group), the student server, the student printers, and the student applications in a container called "Students." You could associate the restricted network access (an attribute) with this container so that these students could access the school's applications and the Internet only during certain hours of the day. Unlike an object, a container can hold multiple objects. Also, a container is a logical construct—that is, a means of organizing other things; it does not represent something real. A container is different from a group because it can hold and apply parameters for many different types of objects, not only users.

Another concept you should understand when working with NOS directories is the idea of a tree. A **tree** is a logical representation of multiple, hierarchical levels in a directory. The term "tree" is drawn from the fact that the whole structure shares a common starting point (the root) and from that point extends branches (or containers), which may

extend additional branches, and so on. Objects are the last items in the hierarchy connected to the branches (and in fact, are sometimes called "leaf objects"). Figure 8-3 depicts a simple directory tree.

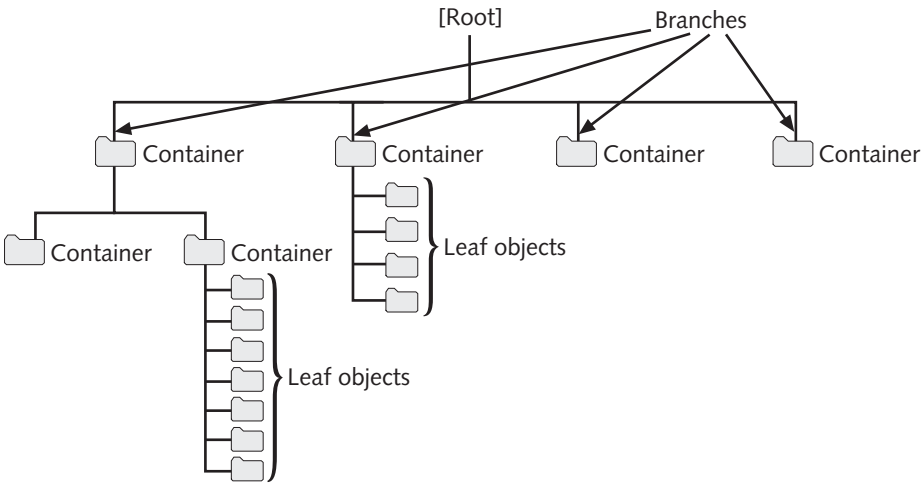

Figure 8-3 A directory tree

Before you install a network operating system, be sure to plan the directory tree with current and future needs in mind. For example, suppose you work at a new manufacturing firm called Circuits Now that produces high-quality, inexpensive circuit boards. You might decide to create a simple tree that branches into three containers: users, printers, and computers. But if Circuits Now plans to open new manufacturing facilities sometime in the future (for instance, one devoted to making memory chips and another for transistors), you might want to call the first container in the tree "circuit boards." This would separate the existing circuit board business from the new businesses, which would employ different people and require different resources. Figure 8-4 shows both possible trees.

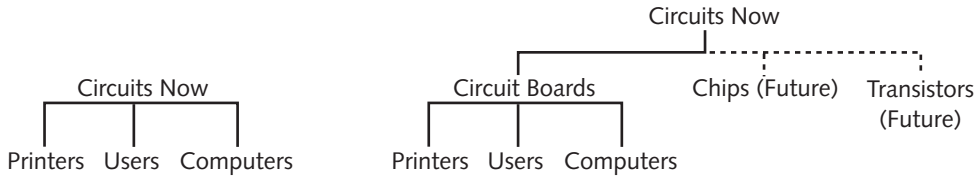

Figure 8-4 Two possible trees for the same organization

Directory trees are very flexible, and as a result, may seem complicated. Chances are that you will enter an organization that has already established its tree, and you will need to understand the logic of that tree to perform your tasks. Later in this chapter and in the two following chapters you will learn more about each NOS's directory and the differences between them.

File Systems

The term **file system** refers to an operating system's method of organizing, managing, and accessing its files through logical structures and software routines. Be careful not to confuse file systems with directories. A file system is different from a directory of files (such as you would view on a PC) because a file system interacts with the operating system. Its purpose is to ensure that the operating system can find requested files on the hard disk. A directory, on the other hand, logically organizes files so that a user can find them on a hard disk. In fact, two files located in the same directory on a computer may be in two different places in the file system.

To administer a network, you should understand the file system (or files systems) used by your NOS. Different file systems not only organize files differently, but also have different file access, speed, compression, defragmentation, and security characteristics. You will need to know which file system your NOS can use and, also, which file systems are compatible with each other. If you completed coursework for the A+ certification, you should already be familiar with the general file systems discussed in the following sections. File systems specific to Windows 2000 Server, NetWare 5.x, and UNIX are discussed elsewhere in this chapter and in Chapters 9 and 10.

FAT (File Allocation Table)

FAT (File Allocation Table) is the original PC file system that was designed in the 1970s to support floppy disks and, later, hard disks. To understand FAT, you must first understand the distribution of data on a disk. Disks are divided into allocation units (also known as clusters). Each allocation unit represents a small portion of the disk's space; depending on your operating system, the allocation unit's size may or may not be customizable. A number of allocation units combine to form a partition. The FAT, a hidden file positioned at the beginning of a partition, keeps track of used and unused allocation units on that partition. The FAT also contains information about the files within each directory, as well as the size of files, their names, and the times that they were created and updated.

When part of a disk uses the FAT method of tracking files, that portion of the disk is called a "FAT partition."

FAT16

The original version of the FAT file system, designed for floppy disks, allows for allocation units that are 12 bits in size. Another version of FAT, designed for computer hard disks, uses 16-bit allocation units. This version of the FAT file system is known as **FAT16**. FAT16 was the standard file system for early DOS- and Windows-based computers. But FAT16 has proved inadequate for most modern operating systems because of its partition size limitations, naming limitations, fragmentation, security, and speed issues. Some sig-

nificant FAT16 characteristics are listed below. (Note the differences between Microsoft's version of FAT16 and the standard FAT16.)

- A FAT16 partition or file cannot exceed 2 GB (when FAT16 is used with the Windows 2000 file system, its maximum size is 4 GB).
- FAT16 uses 16-bit fields to store file size information.
- FAT16 (without additional utilities) supports only filenames with a maximum of eight characters in the name and three characters in the extension.
- FAT16 categorizes files on a disk as Read (a user can read the file), Write (a user can modify or create the file), System (only the operating system can read or write the file), Hidden (a user cannot see the file on the drive without explicitly searching for hidden files), or Archive (used to indicate whether the file has recently been backed up). The term "file attributes" refers to these settings. For example, one file might have a Read file attribute, while another might have a Write file attribute.
- A FAT16 drive stores data in noncontiguous blocks and uses links between fragments to ensure that data belonging to the same file, for example, can be pieced together when the file is requested by the operating system. This approach is unreliable and inefficient, and it may cause corruption.
- Because of FAT16's low overhead, it can write data to a hard disk very quickly.

FAT32

The FAT16 file system was enhanced in the mid-1990s to accommodate longer filenames and permit faster data access via 32-bit addressing. This version of FAT, called **FAT32**, retains some features of the original FAT, such as the Read, Write, System, Hidden, and Archive file attributes. But in contrast to FAT16, FAT32 reduces the maximum size limit file clusters so that space on a disk is used more efficiently. In some cases, FAT32 can conserve as much as 15% of the space that would be required for the same number of files on a FAT16 partition. These and other FAT32 characteristics are listed below:

- FAT32 uses 28-bit fields to store file size information (4 of the 32 bits are reserved).
- FAT32 supports long filenames.
- FAT32 theoretically supports partitions up to 2 Terabytes in size (in Windows 2000, however, the maximum FAT32 partition size is 32 Megabytes).
- Unlike FAT16 partitions, FAT32 partitions can be easily resized without damaging data.
- FAT32 provides greater security than FAT16. For these reasons, FAT32 is preferred over FAT16 for modern operating systems.
- FAT32 is supported by Windows 9x, Windows Me, and Windows 2000.

HPFS (High-Performance File System)

HPFS (High-Performance File System) is a file system originally designed for IBM's OS/2 operating system that offers greater efficiency and reliability than FAT. HPFS organizes data in contiguous blocks, allows data to wait in memory if the processor is too busy to accept it, and assigns information about other data on the disk to each block of data. Collectively, all of these measures enhance HPFS's speed. HPFS also supports extended attributes. In this context, the term **extended attributes** refers to the attributes beyond the basic Read, Write, System, Hidden, and Archive attributes supported by FAT. For example, HPFS provides information about file history, the application to which the file belongs, executable code, icons, and files that depend on other files to function properly. Because it uses 32-bit fields to store file size information, HPFS can handle larger disk sizes than FAT can. HPFS also supports long filenames.

Sharing Applications

As you have learned, one of the significant advantages of the client/server architecture is the ability to share resources, thereby reducing costs and the time required to manage the resources. Along with data storage, applications are an important shared resource. In this section you will learn how a network operating system enables clients to share applications.

Shared applications are often installed on a file server that is specifically designed to run applications. In a smaller organization, however, they may be installed on the same server that provides other functions, such as Internet, security, and remote access services. As a network administrator, you must be sure to purchase a license for the application that allows it to be shared among clients. In other words, you cannot legally purchase one licensed copy of Microsoft Office, install it on a server, and allow all of your 1000 users to share it. Most software vendors sell client licenses for multiple users, which are still much less expensive than purchasing a separate software package for each user. For some applications, you can purchase a **site license**, which (for a fixed price) allows any number of users in one location to legally access that application.

Before selecting an application to run on your network, make sure it is supported by your NOS. You can determine which applications an NOS supports by reading the NOS's documentation or consulting the vendor's Web site. Microsoft, for example, allows you to check whether your applications are Windows 2000 Server-compliant using a search form at *www.microsoft.com/windows2000/professional/howtobuy/upgrading/compat/search/software.asp*.

Once you have purchased the appropriate type and number of licenses, you are ready to install the application on a server. Before doing so, however, you should make sure your server has enough hard disk, memory, and processing power to run the application. Then follow the software manufacturer's guidelines for a server installation. Depending on the application, this process may be the same as installing the application on a workstation or much different.

Once you have installed the software on a server, you are ready to make it available to clients. Through the network operating system, you must assign users rights to the directories where the application's files are installed. Users will at least need rights to access and read files in those directories. For some applications, you may also need to give users rights to create, erase, or modify files associated with the application. For example, a database program may create a small temporary file on the server when a user launches the program to indicate to other potential users that the database is open. If this is the case, users must have rights to create files in the directory where this temporary file is kept. An application's installation guidelines will indicate the rights you need to assign users for each of the application's directories.

Next you will need to provide users a way of accessing the application. On Windows-based or Macintosh clients, you can create an icon on the user's desktop that is associated with the application file. When the user double-clicks the icon, her client software will issue the request to the server to open the application. In response, the network operating system will send a part of the program to her workstation, where it will be held in RAM. This allows the user to interact with the program quickly, without having to relay every command over the network to the server. As the user works with the application, the amount of processing that occurs on her workstation versus the amount of processing that the server handles will vary according to the network architecture. In the preceding example, where a client launches the application directly from a file on the server, the client performs nearly all of the processing. In the case of 3-tier applications, the processing burden is shared between the client and server. And in the case of remote access application sharing, the server performs all of the processing and sends only screen images over the network to the client.

You may wonder how an application can operate efficiently or accurately when multiple users are simultaneously accessing its files. After all, an application's program file is a single resource. If two or more network users double-click their application icon simultaneously, how does the application know which client to respond to? In fact, the network operating system is responsible for arbitrating access to these files. In the case of multiple users simultaneously launching a network application from their desktop icons, the network operating system will respond to one request, then the next, then the next, each time issuing a copy of the program to the client's RAM. In this way, each client is technically working with a separate instance of the application.

Shared access becomes more problematic when multiple users are simultaneously accessing the same data files as well as the same program files. For example, an online auction site accepts bids on many items from many Internet users. Imagine that an auction is nearing a close with three users simultaneously bidding on the same stereo. How does the auction site's database accept bid data for that stereo from multiple sources? One solution to this problem is middleware. The three Internet bidders cannot directly modify the database, located on the auction site's server. Instead, a middleware program runs on the server to accept data from the clients. If the database is not busy, the middleware passes a bid to the database. If the database is busy (or open), the middleware queues the

bids (forces them to wait) until the database is ready to rewrite its existing data, then passes one bid, then another, and another, to the database until its queue is empty. In this way, only one client's data can be written to the database at any point in time.

Sharing Printers

You have learned that sharing peripherals, such as printers, can increase the efficiency of managing resources and reduce costs for an organization. In this section you will learn how networks enable clients to share printers. Sharing other peripheral devices, such as fax machines, works in a similar manner.

In most cases an organization will designate a server as the print server—that is, as the server in charge of managing print services. A printer may be directly attached to the print server or more likely, be attached to the network in a location convenient for the users. In other cases, shared printers may be attached to networked workstations. In order for these printers to be accessible, the workstation must be turned on and functioning properly. Figure 8-5 depicts multiple ways to share printers on a network.

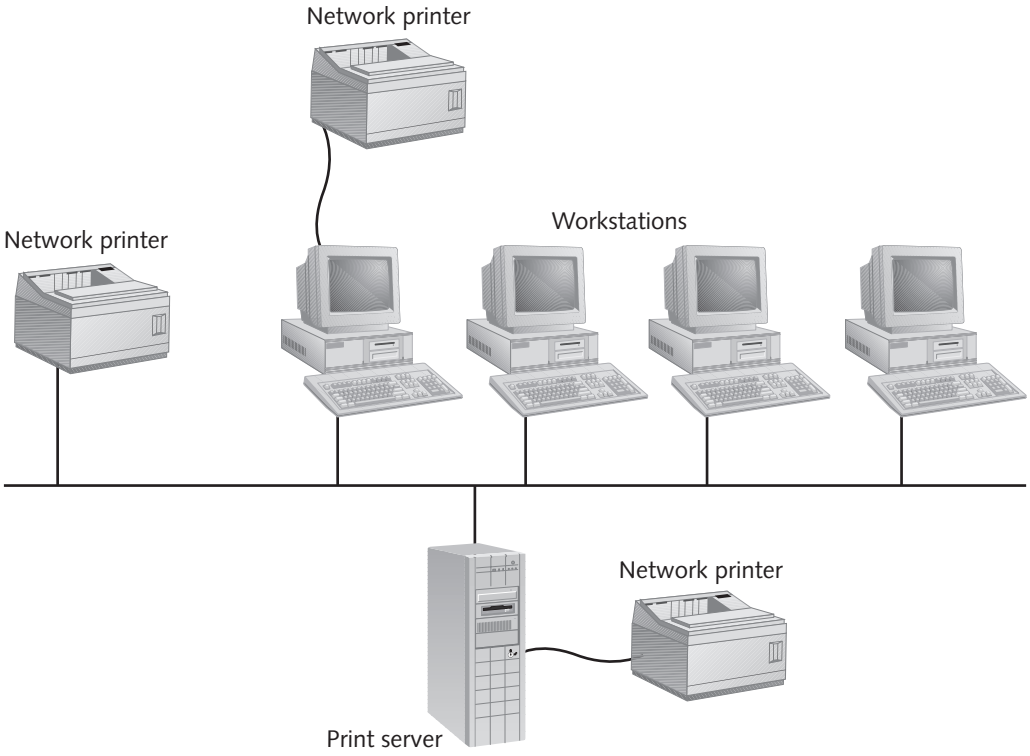

Figure 8-5 Shared printers on a network

Once the printer is physically connected to the network, it needs to be recognized and managed by the NOS before users can access it. Different NOSs have different interfaces for managing printers, but all NOSs can:

- Create an object that identifies the printer to the rest of the network
- Assign the printer a unique name
- Install drivers associated with the printer
- Modify printer attributes, such as location and printing preferences
- Establish or limit access to the printer
- Remotely test and monitor printer functionality
- Update and maintain printer drivers

As a network administrator, you should establish a plan for naming printers before you install them. Since the names you assign the printers will appear in lists of printers available to clients, you should choose names that users can easily decipher. For example, an HP LaserJet 5M in the Engineering Department may be called "ENG_HP5M," or an HP LaserJet 6P in the southwest corner of the building may be called "HP6P_SOUTHWEST." Whatever convention you choose, remain consistent to avoid user confusion and to make your own job easier.

NOSs provide special interfaces for creating new printer objects and assigning them attributes. In Windows 2000 Server, an Add Printer Wizard takes you through the printer creation process step by step. In NetWare 5.x you begin by choosing to create a new object; then a series of menu options leads you through the process, beginning with a printer identification screen, as shown in Figure 8-6. Note the type of attributes this screen allows you to specify.

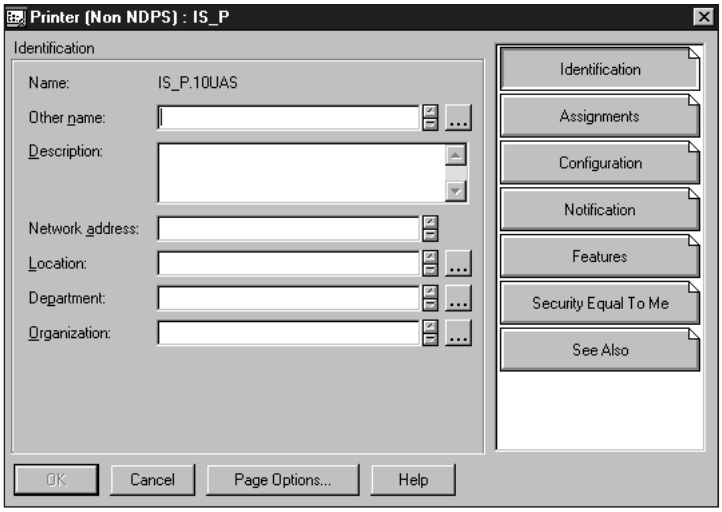

Figure 8-6 NetWare printer identification screen

As you create the new printer, the NOS will require you to install a printer driver, unless one is already installed on the server. This makes the printer's device driver files accessible to users who wish to send jobs to that printer. Before users can access the printer, however, you must ensure that they have proper rights to the printer's queue. The **printer queue** (or share, as it is known in Microsoft terminology) is a logical representation of the printer's input and output. That is, a queue does not physically exist, but rather acts as a sort of "virtual in box and out box" for the printer. When a user wants to print a document, he sends it to the printer queue. To send it to the printer queue, he must have rights to access that queue. As with shared data, the rights to shared printers can vary. Users may have minimal privileges, which allow them to simply send jobs to the printer, or they may have advanced privileges, which allow them to change the priority of print jobs in the queue, or even (in the case of an administrator) change the name of the queue.

Networked printers appear as icons in the Printers folder on Windows and Macintosh workstations, just as local printers would appear. Once they have found a networked printer, users can send documents to that printer just as they would send documents to a local printer. When a user chooses to print, the client redirector determines whether the request should be transmitted to the network or remain at the workstation. On the network, the user's request gets passed to the print server, which puts the job into the appropriate printer queue for transmission to the printer. Figure 8-7 depicts this process.

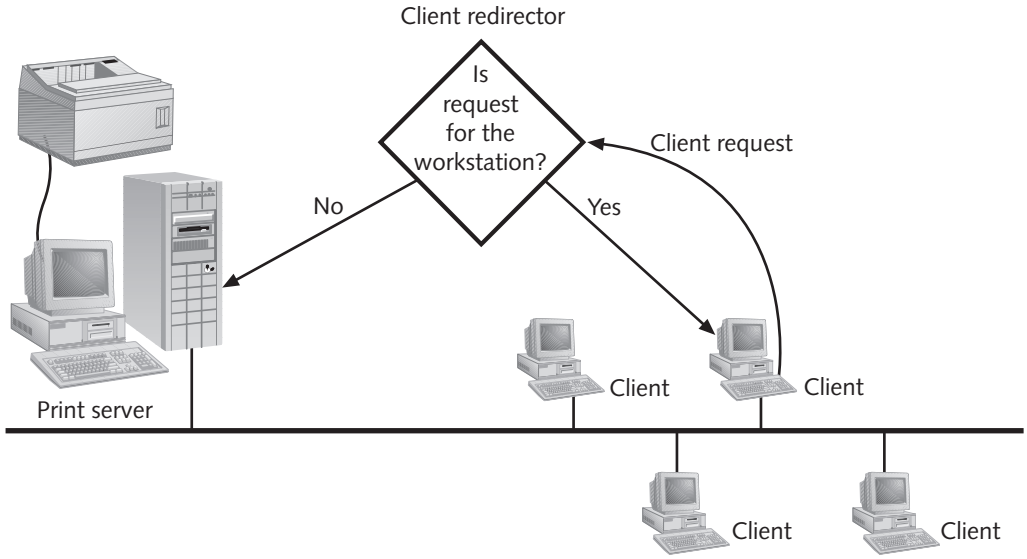

Figure 8-7 Client issuing a job to a networked printer

Managing System Resources

Because a server's system resources (for example, memory and CPU) are limited and are required by multiple users, it is important to make the best use of them. Modern network operating systems have capabilities that maximize the use of a server's memory, CPU, bus, and hard disk. The result is that a server can accommodate more client requests faster—thus improving overall network performance. In the following sections, you will learn about some NOS techniques for managing a server's resources.

Memory

From working with PCs, you may be familiar with the technique of using virtual memory to boost the total memory available to a system. Servers can use both physical and virtual memory, too, as this section describes.

Before learning about virtual memory, you should understand physical memory. The term **physical memory** refers to the (RAM) chips installed on the computer's system board that provide dedicated memory to that machine. The amount of physical memory required by your server varies depending on the tasks that it performs. For example, the minimum amount of physical memory required to run a Windows 2000 Server is 256 MB. However, if you intend to run file and print sharing, Internet, and remote access services on one server, you will need additional physical memory. (When deciding on the appropriate amount of physical memory for your server, remember that the ability to process instructions also depends on processing speed.)

Another type of memory may be logically carved out of space on the hard disk for temporary use. In this arrangement, both the space on the hard disk and the RAM together form **virtual memory**. Virtual memory is stored on the hard disk as a **page file** (or **swap file**), the use of which is managed by the operating system. Each time the system exceeds its available RAM, blocks of information, called pages, are moved out of RAM and into virtual memory on disk. This technique is called **paging**. When the processor requires the information moved to the page file, the blocks are moved back from virtual memory into RAM.

Virtual memory is both a blessing and a curse. On the one hand, if your server has plenty of hard disk space, you can use virtual memory to easily expand the memory available to server applications. This is a great advantage when a process temporarily needs more memory than the physical memory can provide. Virtual memory is typically engaged by default; it requires no user or administrator intervention and is accessed without the clients' knowledge. (However, as a network administrator, you can modify the amount of hard disk space available for virtual memory). On the other hand, using virtual memory slows operations, because accessing a hard disk takes longer than accessing physical memory. Therefore, an excessive reliance on virtual memory will cost you in terms of performance.

Multitasking

Another technique that helps servers use their system resources more efficiently is multitasking. **Multitasking** is the ability of a processor to perform many different operations in a brief period of time. If you have opened multiple programs simultaneously on a desktop computer, you have taken advantage of your operating system's multitasking capability.

All of the major NOSs can perform multiple tasks at one time. If they couldn't, network performance would be considerably slower, since busy servers are continually receiving and responding to multiple requests. However, NOSs differ in their multitasking abilities.

In NetWare, UNIX, and Windows 2000 Server, the server actually performs one task at a time, allowing one program to use the processor for a certain period of time, and then suspending that program to allow another program to use the processor. Thus, each program has to take turns loading and running. Because no two tasks are ever actually performed at one time, this capability is not considered true multitasking; instead, it is referred to as **preemptive multitasking**. Preemptive multitasking happens so quickly, however, that the average user could probably not distinguish between it and true multitasking.

Multiprocessing

Before you learn about the next method of managing system resources, you need to understand the terms used when discussing data processing. A **process** is a routine of sequential instructions that runs until it has achieved its goal. A word-processing program's executable file is an example of a process. A **thread** is a self-contained, well-defined task within a process. A process may contain many threads, each of which may run independently of the

others. All processes have at least one—the main thread. For example, to eliminate the waiting time when you save a file in your word processor, the programmer who wrote the word-processor program might have chosen to implement the file save operation as a separate thread. That is, the part of the program that implements the file save operation executes in a thread that is independent of the main thread. This independent execution allows you to continue typing while the document is being written to the disk.

On systems with only one processor, only one thread can be handled at any time. Thus, if a number of programs are running simultaneously, no matter how fast the processor, a number of processes and threads will be left to await execution. Using multiple processors allows different threads to run on different processors. The support and use of multiple processors to handle multiple threads is known as **multiprocessing**. Multiprocessing is often used on servers as a technique to improve response time. To take advantage of more than one processor on a computer, its operating system must be capable of multiprocessing. For example, Windows 2000 Server supports the use of four processors.

Multiprocessing splits tasks among more than one processor to expedite the completion of any single instruction. To understand this concept, think of a busy metropolitan freeway during rush hour. If five lanes are available for traffic, drivers can pick any lane—preferably the fastest lane—to get home as soon as possible. If traffic in one lane slows, drivers may choose another, less congested lane. This ability to move from lane to lane allows all traffic to move faster. If the same amount of traffic had to pass through only one lane, everyone would go slower and get home later. In the same way, multiple processors can handle more instructions more rapidly than a single processor could.

NetWare 5.x and Windows 2000 Server support a special type of multiprocessing called **symmetric multiprocessing**, which splits all operations equally among two or more processors. Another type of multiprocessing, **asymmetric multiprocessing**, assigns each subtask to a specific processor. Continuing the freeway analogy, asymmetric multiprocessing would decree that all semi trucks must use the far right lane, all pickup trucks must use the second to the right lane, all compact cars must use the far left lane, and so on. The efficiency of each multiprocessing model is open to debate, but, in general, symmetric processing completes operations more quickly because the processing load is more evenly distributed.

Multiprocessing offers a great advantage to servers with high CPU usage—that is, servers that perform numerous tasks simultaneously. If an organization uses its server merely for file and print sharing, however, multiple processors may not be necessary. You should carefully assess your processing needs before purchasing a server with multiple processors. Some processing bottlenecks are not actually caused by the processor—but rather by the time it takes to access the server's hard disks or by problems related to cabling or connectivity devices. Determining the source of network performance degradation can be an art, and you will learn more about this practice in Chapter 12.

INTRODUCTION TO WINDOWS 2000 SERVER

Windows 2000 Server is the latest version of Microsoft's network operating system. It was released in 1999 but was under development for five years before that. Windows 2000 Server serves as a redesign and enhancement of its predecessor, Windows NT Server. Windows NT Server was a popular network operating system known for its intuitive graphical user interface, multitasking capabilities, and compatibility with a huge array of applications. A **graphical user interface** (**GUI**; pronounced "gooey") is a pictorial representation of computer functions that, in the case of network operating systems, enables administrators to manage files, users, groups, security, printers, and so on. When Windows NT Server was commercially released in 1993, it was the first network operating system based entirely on a GUI, making network administration easier than ever before. Prior to Windows NT, the only option Microsoft provided for sharing resources between Windows-based workstations was Windows for Workgroups, which employed a peer-to-peer network model.

Windows 2000 Server carries on many of the advantages of Windows NT Server, plus provides additional features and capabilities. Some benefits of the Windows 2000 Server NOS include:

- An advanced system of organizing and managing network objects, called Active Directory
- Multiple, integrated Web services with an easy to use administrator interface
- Support for a great deal of RAM and multiple processors
- Support for multiple, modern protocols and security standards
- Excellent integration with other network operating systems
- Simple centralized management of multiple clients
- Flexible, customizable network management interface

With Windows 2000, Microsoft in fact released three different, but related NOSs: Windows 2000 Server, Windows 2000 Advanced Server, and Windows 2000 Datacenter Server. Windows 2000 Advanced Server offers the same benefits and includes the same features as Windows 2000 Server, but adds support for **clustering**, a method of connecting multiple servers to enable resource sharing and load balancing between them. You will learn more about clustering in Chapter 14. Windows 2000 Advanced Server also supports servers with up to eight processors and up to 8 GB of RAM, while Windows 2000 Server supports up to four processors and up to 4 GB of RAM. In short, Windows 2000 Advanced Server is generally suitable for a larger enterprise. Windows 2000 Datacenter Server is designed for environments that make heavy use of databases and data manipulation. Windows 2000 Datacenter Server supports up to 32 processors and 64 GB of RAM.

In addition to these options, Microsoft also offers Windows 2000 Professional, a desktop operating system that can also accept some client connections. You may wonder how Windows 2000 Server differs from Windows 2000 Professional. In general, Windows 2000 Server provides more services specifically targeted to networks. While Windows 2000 Professional can accept up to 10 client connections, Windows 2000 Server can accept nearly unlimited client connections. Windows 2000 Server also provides network security, storage, Web, and management features that are not included in Windows 2000 Professional.

This chapter gives a broad overview of how Windows 2000 Server (the basic version) fits into a network environment. It also provides other information necessary to qualify for Net+ certification. It does not attempt to give exhaustive details of the process of installing, maintaining, or optimizing Windows 2000 networks. For this in-depth knowledge (particularly if you plan to pursue MCSE certification), you should invest in books devoted to Windows 2000 Server, such as Course Technology's *MCSE Guide to Microsoft Windows 2000 Server*, ISBN 0-619-01517-9.

Why Choose Windows 2000 Server?

Windows 2000 Server is a popular network operating system because it addresses most of a network administrator's needs very well. Microsoft is, of course, a well-established vendor that wields its size and influence to ensure that other programs will be compatible with its systems. Its large market share also guarantees that technical support—whether through Microsoft, private developer groups, or third-party newsgroups—are readily available. If you become MCSE-certified, you will be eligible to receive enhanced support directly from Microsoft. This enhanced support (including a series of CDs) will help you solve problems more quickly and accurately. Because Windows 2000 is so widely used, you can also search newsgroups on the Web and will probably find someone who has encountered and solved a problem like yours.

Windows 2000 supports any type of topology or protocol that you are likely to run on a LAN. This efficient network operating system takes advantage of multiple processors. Its multitasking capabilities allow server-based processes (for example, retrieving a file, sending a command to a networked printer, or authorizing a user to log on) to share CPU resources. Also, its customizable, graphical administrative interface called the **Microsoft Management Console (MMC)** makes Windows 2000 Server a simple operating system to manage. Thus, technical staff members who are unfamiliar with the system can quickly learn how to perform routine maintenance and operation activities. Windows 2000 Server also provides excellent security features, which are highly customizable and easily managed from the server console.

One potential drawback to using Windows 2000 Server is its performance. In independent benchmark tests, Windows 2000 Server has been found to read and write data somewhat slower than NetWare 5.x and certain versions of UNIX. On the other hand,

performance greatly depends on the type of routines and commands tested. Windows 2000 Server's broad range of features overshadows any perceived performance disadvantage for most environments. Its ease of centralized management, Web integration, storage management, and built-in system monitoring tools are unique advantages over the other major NOSs.

Since its release in 1999, Windows 2000 Server has become the NOS of choice for a wide variety of environments. If you have used the Web to research a topic, conduct online banking transactions, reserve a hotel, or purchase gifts, chances are you have connected to Windows 2000 Server. Windows 2000 Server is also used in hospitals, schools, government offices, warehouses, manufacturing plants, utilities, stores, and media outlets. Because of Windows 2000's cross-industry acceptance, understanding this NOS will likely provide you with a valuable career skill.

Windows 2000 Server Hardware

You have learned that servers generally require more processing power, more memory, and more hard disk space than workstation machines do. In addition, servers may contain redundant components for fault tolerance, self-monitoring firmware, multiple processors and NICs, or peripherals other than the common CD-ROM and floppy disk drives. The type of servers you choose for your network will depend partly on your network operating system. As you learned earlier, each network operating system demands specific requirements in terms of server hardware.

An important resource for determining what kind of Windows 2000 hardware to purchase is Microsoft's Hardware Compatibility List. The **Hardware Compatibility List (HCL)** lists all computer components proven to be compatible with Windows 2000 Server. The HCL appears on the same CD-ROM as your Windows 2000 Server software. If you don't find a hardware component on the HCL that shipped with your software, you can look it up on Microsoft's Web site. At the time of this writing, Microsoft's searchable hardware compatibility list could be accessed from the following Web site: *www.microsoft.com/windows2000/server/howtobuy/upgrading/compat/default.asp*. You should always consult this list before buying new hardware. Although hardware that is *not* listed on the HCL may work with Windows 2000 Server, Microsoft's technical support won't help you solve problems related to such hardware.

Table 8-2 lists Microsoft's minimum server requirements for Windows 2000 Server.

Table 8-2 Minimum hardware requirements for Windows 2000 Server

Component	Requirement
Processor	133 MHz or higher Pentium or Pentium-compatible processor. Windows 2000 Server supports up to four CPUs in one server; however, Windows 2000 Advanced Server can support up to eight CPUs and Windows 2000 Datacenter Server can support up to 32 CPUs
Memory	256 megabytes (MB) of RAM is the recommended minimum (but a 128 MB minimum is supported). A computer running Windows 2000 Server may hold a maximum of 4 gigabytes (GB) of memory
Hard disk drive	A hard drive supported by Windows 2000 (as specified in the HCL) with a minimum of 1 GB of free space available for system files (2 GB recommended)
NIC	Although a NIC is not required to install the Windows 2000 Server NOS, it is required to connect to a network. Use a NIC found on the HCL. More than one NIC can be supported
CD-ROM	A CD-ROM drive found on the HCL is required unless the installation will take place over the network
Pointing device	A mouse or other pointing device found on the HCL
Floppy disk drive	Not required

Minimum requirements specify the *least* amount of RAM, hard disk space, and processing power you must have to run the network operating system. Your applications and performance demands, however, may require more resources. Some of the minimum requirements listed in Table 8-2 (for example, the 133 MHz Pentium processor) may apply to the smallest test system—not a realistic networking environment. Be sure to calculate the optimal configuration for your network's server based on your environment's needs before you purchase new hardware. For instance, you should make a list of every application and utility you expect the server to run in addition to the NOS. Then look up the processor, memory, and hard disk requirements for each of those programs and estimate how significantly their requirements will affect your server's overall hardware requirements. It is easier and more efficient to perform an analysis before you install the server than to add hardware after your server is up and running.

A Closer Look at the Windows 2000 Server Network Operating System

By now you should understand some of the features that are important to all network operating systems. You should also have a sense of the type of organization that might choose Windows 2000 Server as its preferred NOS. In the next sections, you will learn specifically how Windows 2000 Server manages its system resources, data files, and network objects.

Windows 2000 Server Memory Model

Earlier you learned that Windows 2000 Server can make use of up to 4 processors and, further, that it employs a type of multiprocessing called symmetric multiprocessing. You also learned that Windows 2000 Server can make use of virtual memory. This section provides more information on how Windows 2000 optimizes its use of a server's memory to perform many complex tasks simultaneously.

The Windows 2000 Server memory model uses a 32-bit addressing scheme. To appreciate the advantages of this feature, you may want to review the material on addressing in Chapter 6 (in the discussion of bus-adapter NICs). Essentially, the larger the addressing size, the more efficiently instructions can be processed. Microsoft's original network operating system used a 16-bit addressing scheme, half the size of that supported by Windows 2000 Server.

The Windows 2000 Server memory model also assigns each application (or process) its own 32-bit memory area. This memory area is a logical subdivision of the entire amount of memory available to the server. Assigning separate areas to processes helps prevent one process from interfering with another's operations, even though the processes are running simultaneously. The technique is analogous to a company giving each employee a car to commute to work. You can imagine the disadvantage of this approach: It is less efficient than supplying employees with a few commuter vans that can transport 10 people to work at the same time. The risk of losing employees to injuries suffered in an accident is lower, however, because a commuter van accident affects more employees than an accident involving an individual car driver. Similarly, if each application uses the same memory area, one misbehaving process can take down all applications running in that memory area. On the other hand, if each application uses a separate memory area, each one can harm only itself. As in the car-van scenario, using separate memory spaces for each application is, however, less efficient.

Another important feature of the Windows 2000 Server memory model is that it allows you to install more physical memory on the server than previous versions of Windows did, which in turn means that the server can process more instructions faster.

Finally, as you have learned, Windows 2000 Server can make use of virtual memory. To find out how much virtual memory your Windows 2000 server uses, open the Control Panel, double-click the System icon, click Advanced, click Performance Options, and then click the Change button. If multiple drives are listed in the Drive box, choose the drive that contains your page file. The Paging file size indicates how much hard disk space is available for virtual memory, as pictured in Figure 8-8. If you suspect that your server's processing is being degraded because it relies on virtual memory too often, you should invest in additional physical memory (RAM).

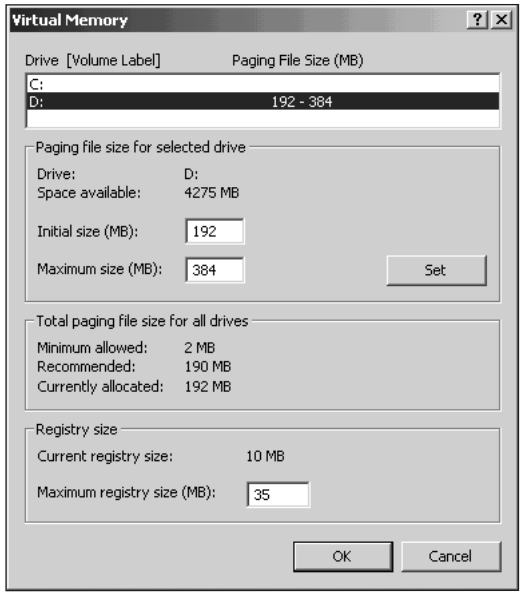

Figure 8-8 Viewing virtual memory

Windows 2000 File Systems

In addition to the FAT, FAT32, and HPFS file systems you learned about earlier, Windows 2000 Server supports other types of file systems, including CDFS and NTFS. You will learn about both in the following sections. You will also learn when it is most appropriate to use NTFS and FAT32 on your Windows 2000 server.

CDFS (CD-ROM File System) and UDF (Universal Disk Format)

The **CDFS (CD-ROM File System)** is the file system used to read from and write to a CD-ROM disk. Windows 2000 Server supports CDFS so as to allow program installations and CD-ROM file sharing over the network. No intervention is necessary to install or configure the CDFS—it is installed automatically when you install Windows 2000 Server. In addition to CDFS, Windows 2000 Server supports the **Universal Disk Format (UDF)**, which is another file system used on CD-ROMs and digital video disc (DVD) media. DVDs and CD-ROMs can be used to store large quantities of data in a networking environment. However, they are more commonly used to store digital audio and video data, such as movies.

NTFS (New Technology File System)

Microsoft developed **NTFS (New Technology File System)** expressly for its Windows NT platform. With the release of Windows 2000, Microsoft updated NTFS to version 5. Among other things, this upgrade provided NTFS with better support for

other file systems and for newer security techniques. NTFS is reliable and makes it possible to compress files so they take up less space. At the same time, NTFS can handle massive files, and allow fast access to data, programs, and other shared resources. It is used only on Windows NT or Windows 2000 servers (in other words, UNIX and NetWare servers cannot, by default, make use of NTFS). If you are working with Windows 2000 Server, choose NTFS for your server's file system (for reasons discussed later in this section). Therefore, you should familiarize yourself with the following NTFS features:

- NTFS filenames can be a maximum of 255 characters long.
- NTFS stores file size information in 64-bit fields.
- NTFS files or partitions can theoretically be as large as 16 exabytes, (2^{64} bytes).
- NTFS is required for Macintosh connectivity.
- NTFS incorporates sophisticated, customizable compression routines. These compression routines reduce the space taken by files by as much as 40%. A 10 GB database file, for example, could be squeezed into 6 GB of disk space.
- NTFS keeps a log of file system activity to facilitate recovery if a system crash occurs.
- NTFS is required for encryption and advanced access security for files, user accounts, and processes.
- NTFS improves fault tolerance through RAID and system file redundancy.

Before installing Windows 2000 Server, you should decide which file system (or systems) you will use. In general, you need to worry about only the FAT32 and NTFS file systems, because HPFS is not native to Windows 2000 and CDFS and UDF are installed automatically. Although FAT32 improves upon the FAT16 file system and typically appears on Windows 9x workstations, it is not optimal for Windows 2000 servers. Instead, the NTFS file system is preferred because it enables a network administrator to take advantage of the Windows 2000 security and file compression enhancements.

One drawback to using an NTFS partition is that it cannot be read by FAT16, FAT32, or HPFS partitions (unless you employ a third-party utility). However, NTFS partitions can read FAT partitions. You should also be aware that you can convert a FAT drive into an NTFS drive on a Windows 2000 server, but you cannot convert an NTFS drive into a FAT drive.

Typically, due to all the benefits listed above, you will select NTFS whenever you install Windows 2000 Server. (Later in this chapter, you will have the opportunity to plan and execute a Windows 2000 Server installation.) The only instance in which you should not use NTFS is if one of your server's applications is incompatible with this file system.

Microsoft Management Console (MMC)

If you have worked with Windows NT servers, you understand that for each administrative function, the NOS provides a separate tool. Also, each tool has a unique, but similar graphical interface. In Windows 2000 Server, Microsoft has integrated all of the NOS's administrative tools into a single interface called the Microsoft Management Console (MMC). This section provides an overview of MMC, its capabilities, and how you can customize it for your network environment.

An MMC is simply an interface. Its purpose is to gather multiple administrative tools into a convenient console for your network environment. If an MMC doesn't contain the tools you want, you can add or remove administrative tools to suit your situation. The tools you add to the interface are known as **snap-ins**. For example, you may be the network administrator for two servers, one that performs data backup services and another dedicated to Web services, on the same network. On the backup server, your MMC should definitely include the disk management snap-in, which allows you to easily manage the hard disk's volumes and the event viewer snap-in, which allows you to view what processes have run on the server and whether they generated any errors. On the Web server, you might want to install the FrontPage Server Extensions, Internet Information Services, and the Internet Authentication Service (IAS) snap-ins. However, if the first server is only used for data backup, there is no need to add these three Internet-related snap-ins to its MMC. You can create multiple MMCs on multiple servers, or even multiple MMCs on one server.

You can find snap-ins either through an MMC or as separate selections from the Administrative Tools menu.

Before using MMCs for the first time, you must create a custom console by running it for the first time and adding your selections. To do so, click Start, click Run, type `mmc` in the text box in the Run dialog box, and then click OK. The Console1 (MMC) window opens as a window separated into two panes, as shown in Figure 8-9. The left pane lists the administrative tools. The right pane lists specific details for a selected tool.

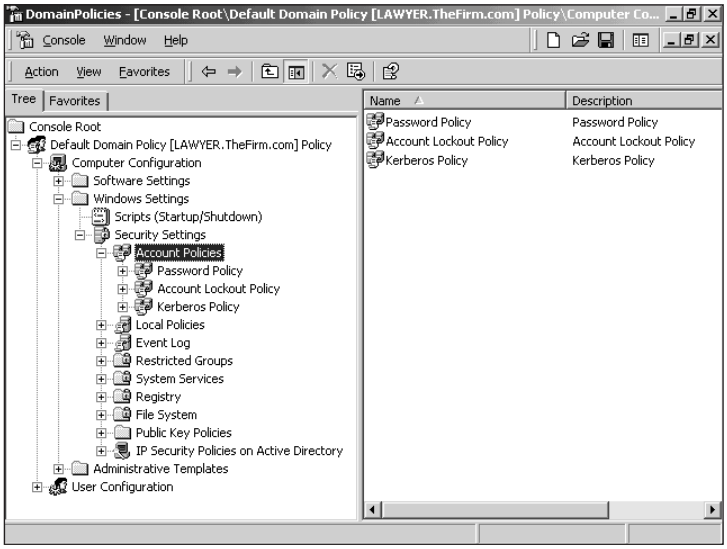

Figure 8-9 MMC window

When you first open the MMC, it does not contain any snap-ins; the panes of its window are empty. You can customize the MMC by adding administrative tools.

To add administrative tools to your MMC interface:

1. Click **Console** in the MMC main menu bar and then click **Add/Remove Snap-in**. The Add/Remove Snap-in dialog box appears, listing the currently installed snap-ins.
2. Click the **Add** button. The Add Standalone Snap-in dialog box appears with a list of available snap-ins.
3. In the Add Standalone Snap-in dialog box, click the tool you want to add to your console and then click **Add**. Continue adding snap-ins until you have chosen all that you want to include in your MMC. (When you add some snap-ins, such as Event Viewer and Device Manager, you will be asked to select the computer that you want the snap-in to manage, and to indicate whether the snap-in should manage the local computer or another computer on the network.)
4. Once you have added all the snap-ins you want, click **Close**. The Add Standalone Snap-in dialog box closes.
5. Click **OK**. The Add/Remove Snap-in dialog box closes and the new tools are added to the MMC. Notice that the left pane of your MMC window now includes the snap-ins you've added.

After you have customized your MMC, you need to save your settings. When you save your settings, you assign a name to the specific console (or administrative interface) that you have just created. Assign the MMC a name that indicates its function. For example, you might create an MMC specifically for managing users and groups and then name that MMC "My User Tool." Later, you can access this same MMC by choosing Start/Programs/Administrative Tools/My User Tool.

MMC can operate in two modes—author mode and user mode. Network administrators who have full permissions on the server typically use author mode, which allows full access for adding, deleting, and modifying snap-ins. However, sometimes an administrator may want to delegate certain network management functions to colleagues, without giving them full permissions on the servers. In such a situation, the administrator can create an MMC that runs in user mode—in other words, that provides limited user privileges. For example, the user might be allowed to view administrative information, but not to modify the snap-ins.

Active Directory

Early in this chapter you learned about NOS directories, the methods for organizing and managing objects on the network. Windows 2000 Server uses a directory service called Active Directory, which was designed especially for Windows 2000 networks. This section provides an overview of how Active Directory is structured and how it uses standard naming conventions to better integrate with other networks. You'll also learn how Active Directory stores information for Windows domains.

Schema

Before installing a NOS, you must have a thorough understanding of how its servers, users, groups, and resources are logically and physically organized and related. This is especially significant if you are planning a new Windows 2000 network.

To begin with, you should understand the foundation of Active Directory's structure. Active Directory is a database that contains records of objects and information about those objects. To be logically organized and easily accessible by multiple types of programs, a database must have definitions for its components. For example, if you design a payroll database for your company's human resources department, you would probably want to include "employee" as a component, and "first name," "last name," "phone number," "address," "salary," and other information associated with an employee. Once you decide how to organize the database and what to include, you must make note of the components and information fields you have created. That way, when a programmer creates a user interface to the database, he will know that he should call the "employee" component when retrieving an employee's record. The same principle applies to Active Directory's objects. A **schema** is the set of definitions of the kinds of objects and information associated with those objects that the database can contain. For example, one type of object is a printer, and one type of information associated with that object is the location of the printer. Thus, "printer," and "location of printer" would be definitions contained within the schema.

Figure 8-10 shows the relationship between Active Directory and a simple user account schema.

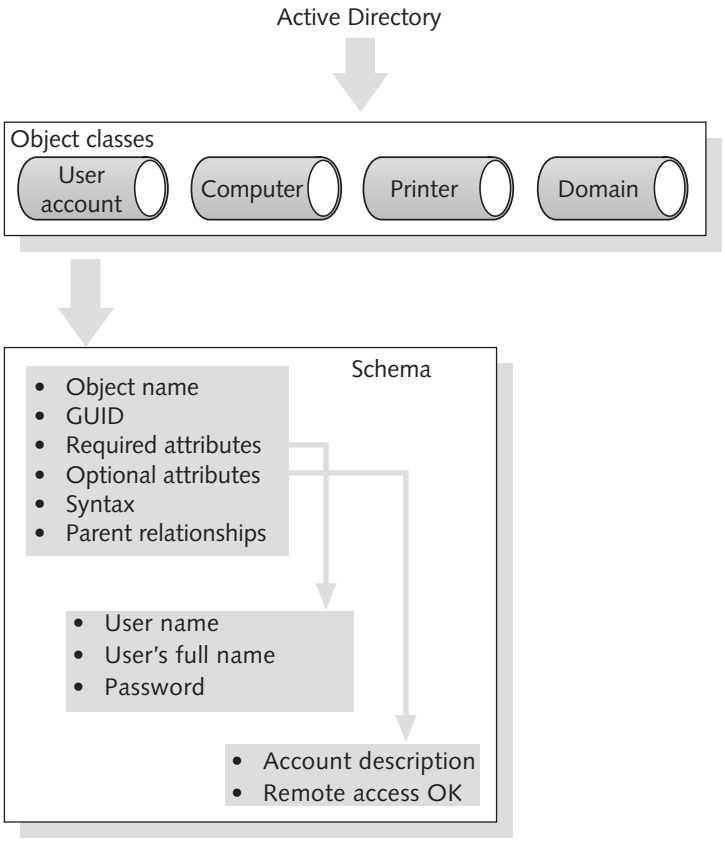

Figure 8-10 Active Directory and a simple user schema

Active Directory's schema may contain two types of definitions: classes and attributes. **Classes** (also known as **object classes**) identify what type of objects can be specified in Active Directory. User account is an example of an object class. Another object class is Printer. An attribute, as you learned before, is a property associated with an object. For example, Home Directory is the name of an attribute associated with the User object, while Location is an attribute associated with the Printer object. Classes are composed of many attributes. When you create an object, you also create a number of attributes that store information about that object. The object class and its attributes are then saved in Active Directory's database.

Workgroups

A Windows 2000 network can be set up in a workgroup model or a domain model. This section describes the workgroup model. In the next section you will learn about the more popular domain model.

A **workgroup** is a group of interconnected computers that share each other's resources without relying on a central server. In other words, a workgroup is a type of peer-to-peer network. Computers in a Windows 2000 workgroup may run either the Windows 2000 Professional or the Windows 2000 Server operating system. Each computer in the workgroup has its own database of user accounts and security privileges, as shown in Figure 8-11.

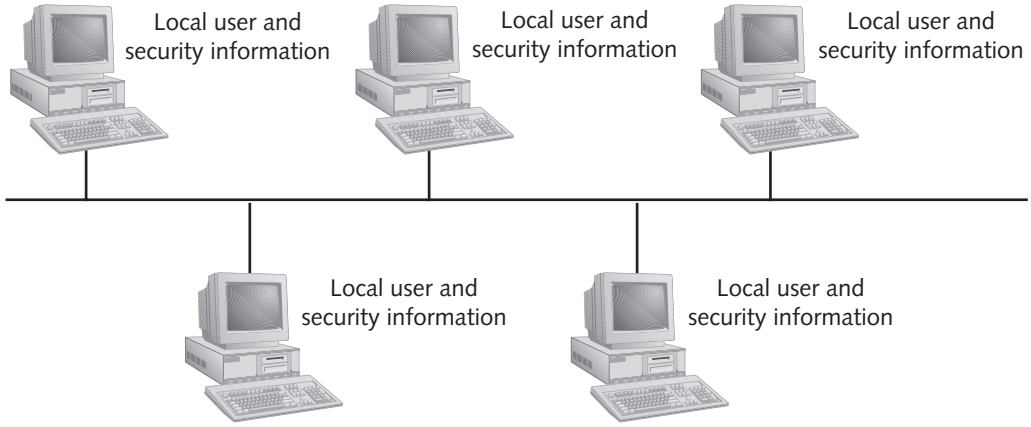

Figure 8-11 A Windows 2000 workgroup network

Because each computer maintains its own database, each user must have a separate account on each computer he wants to access. This decentralized management results in significantly more administration effort than a client/server Windows 2000 network would require. In addition, workgroups are not practical for groups of more than 10 computers. On the other hand, peer-to-peer networks such as a Windows 2000 workgroup are simple to design and implement and may be the best solution for small groups of users who have few security concerns.

Domains

The type of Windows 2000 network that follows the client/server architecture is known as a domain model (meaning that it relies on domains). A **domain** is a group of users, servers, and other resources that share a database of account and security information. The database that domains use to record their objects and attributes is contained within Active Directory (as you will learn, Active Directory contains still more information). Domains are established on a network to make it easier to organize and manage

resources and security. For example, a university might create separate domains for each of the following colleges: Life Sciences, Humanities, Business, Communications, and Engineering. Within the Engineering domain, additional domains such as "Chemical Engineering," "Industrial Engineering," "Electrical Engineering," and "Mechanical Engineering" may be created, as shown in Figure 8-12. In this example, all users, workstations, servers, printers, and other resources within the Engineering domain would share a distinct portion of the Active Directory database.

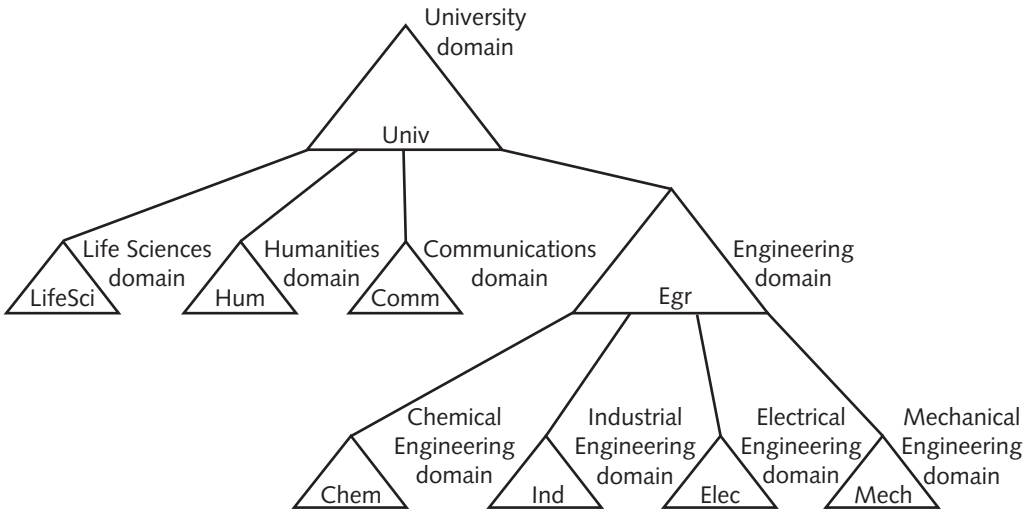

Figure 8-12 Multiple domains in one organization

Keep in mind that a domain is not confined by geographical boundaries. Computers and users belonging to the university's Engineering domain may be located at five different campuses across a state, or even across the globe. No matter where they are located, they obtain their object, resource, and security information from the same database and the same portion of Active Directory.

Depending on the network environment, an administrator can define domains according to function, location, or security requirements. For example, if you worked at a large hospital whose WAN connected the city's central healthcare facility with several satellite clinics, you could create separate domains for each WAN location or you could create separate domains for each clinical department, no matter where they are located. Alternately, you might choose to use only one domain and assign the different locations and specialties to different containers within the domain.

The directory containing information about objects in a domain resides on computers called **domain controllers**. A Windows 2000 network may use multiple domain controllers. In fact, you should use at least two domain controllers on each network so that if one domain controller fails, the other will continue to retain your domains' databases.

Servers on a Windows 2000 network that do not store directory information are known as **member servers**. Because member servers do not contain a database of users and their associated attributes (such as password or permissions to files), member servers cannot authenticate users. Only domain controllers can do that. Every server on a Windows 2000 network is either a domain controller or a member server, as shown in Figure 8-13.

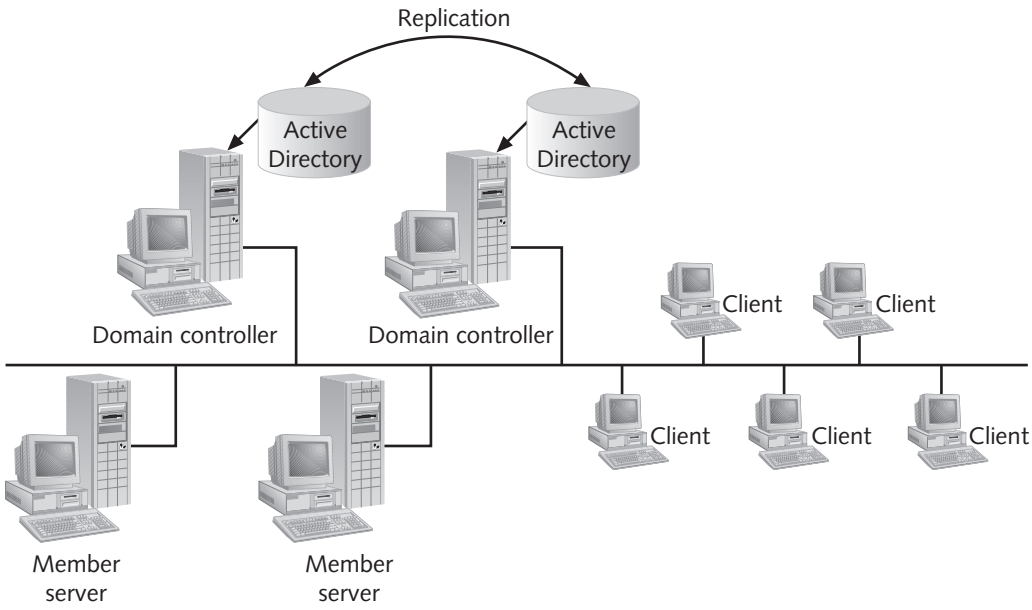

Figure 8-13 A Windows 2000 domain model network

When a network uses multiple domain controllers, a change to the database contained on one domain controller is copied to the databases on other domain controllers so that their databases are always identical. The process of copying directory data to multiple domain controllers is known as **replication**. Replication ensures redundancy so that in case one of the domain controllers fails, another can step in to allow clients to log onto the network, be authenticated, and access resources.

Organizational Units

Earlier you learned that NOSs use the concept of containers to hold multiple objects that have similar characteristics. In Windows 2000, such containers are also known as **organizational units (OUs)**. An OU can contain over 10 million objects. And each OU can contain multiple OUs. For example, suppose you were the network administrator for the university described previously, which has the following domains: Life Sciences, Humanities, Business, Communications, and Engineering. Further, suppose you decided that rather than making additional domains beneath each college, you would group objects according to containers. For the Life Sciences domain, you might create the following organizational units that correspond to the Life Sciences departments: Biology, Geology, Zoology, and Botany. In addition, you might want to create

organizational units for each building that the departments use. For example, "Schroeder" and "Randall" for Biology, "Morehead" and "Kaiser" for Geology, "Randall" and "Arthur" for Zoology, and "Thorne" and "Grieg" for Botany. Figure 8-14 depicts a tree based on this example.

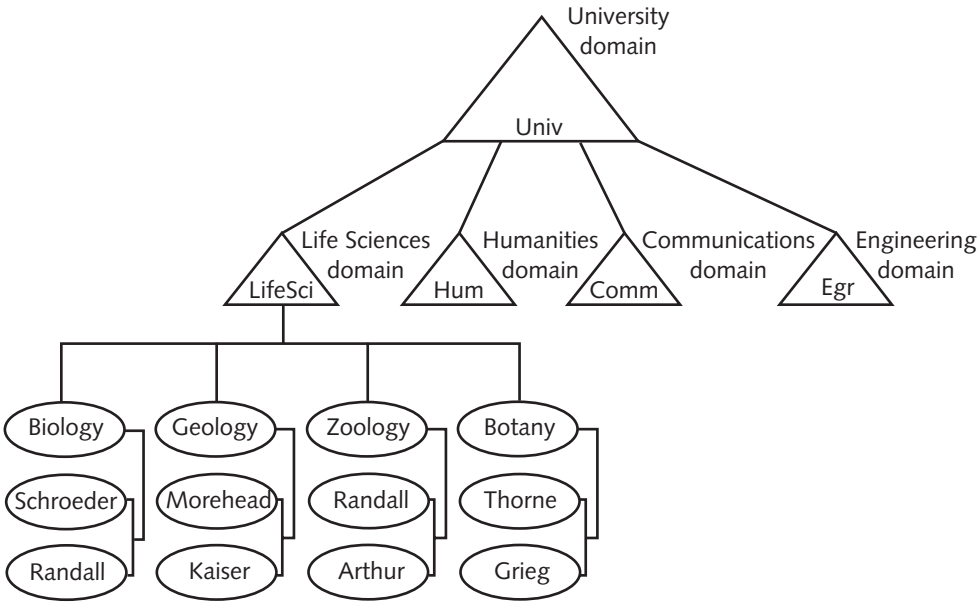

Figure 8-14 A tree with multiple domains and OUs

Trees and Forests

Now that you understand how an NOS directory can contain multiple levels of domains and organizational units, you are ready to learn the structure of the directory that exists above domains. It is common for large organizations to use multiple domains in their Windows 2000 networks. Active Directory organizes multiple domains hierarchically in a **domain tree** (or simply, tree). (Recall that NOS trees were introduced earlier in the chapter. Active Directory's domain tree is an example of a typical NOS tree.) At the base of the Active Directory tree is the **root domain**. From the root domain, **child domains** branch out to separate objects with the same policies, as you saw in Figure 8-12. Underneath the child domains, multiple organizational units branch out to further subdivide the network's systems and objects.

A collection of one or more domain trees is known as a **forest**. All trees in a forest share a common schema. Domains within a forest can communicate, but only domains within the same tree share a common Active Directory database. In addition, objects belonging to different domain trees are named separately, even if they are in the same forest. You will learn more about naming later in this chapter.

Trust Relationships

In order for your network to work efficiently, you must give some thought to the relationships between the domains in a domain tree. The relationship between two domains in which one domain allows another domain to authenticate its users is known as a **trust relationship**. Active Directory supports two types of trust relationships: two-way transitive trusts and explicit one-way trusts. Each child and parent domain within a domain tree and each top-level domain in a forest share a **two-way transitive trust** relationship. This means that a user in domain A is recognized by and can be authenticated by domain B, and vice versa. In addition, a user in domain A may be granted rights to any of the resources managed by domain B, and vice versa.

When a new domain is added to a tree, it immediately shares a two-way trust with the other domains in the tree. These trust relationships allow a user to log onto and be authenticated by a server in any domain within the domain tree. However, this does not necessarily mean that the user has privileges to access any resources in the tree. A user's permissions must be assigned separately for the resources in each different domain. For example, suppose Betty is a research scientist in the Mechanical Engineering Department. Her user account belongs to the Engineering domain at the University. One day, due to construction in her building, she has to temporarily work in an office in the Zoology Department's building across the street. The Zoology Department OU, and all its users and workstations, belong to the Life Sciences domain. When Betty sits down at the computer in her temporary office, she can log onto the network from the Life Sciences domain, which happens to be the default selection on her logon screen. She can do this because the Life Sciences and Engineering domains have a two-way trust. Once she is logged on, she can access all her usual data, programs, and other resources in the Engineering domain. But even though the Life Sciences domain authenticated Betty, she will not automatically have privileges for the resources in the Life Sciences domain. For example, she can retrieve her research reports from the Mechanical Engineering Department's server, but unless a network administrator grants her rights to access the Zoology Department's printer, she cannot print the document to the networked printer outside her temporary office.

Figure 8-15 depicts the concept of a two-way trust between domains in a tree.

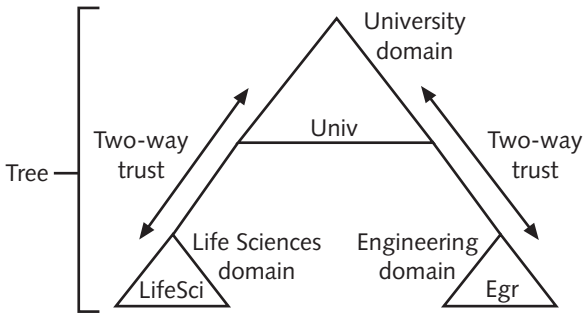

Figure 8-15 Two-way trusts between domains in a tree

The second type of trust relationship supported by Active Directory is an **explicit one-way trust**. In this scenario, two domains that are not part of the same tree are assigned a trust relationship. The explicit one-way trust does not apply to other domains in the tree, however. Figure 8-16 shows how an explicit one-way trust can enable domains from different trees to share resources. In this figure, notice that the Engineering domain in the University tree and the Research domain in the Science Corporation tree share a one-way trust. However, this trust does not apply to parent or child domains associated with the Engineering or Research domains. In other words, the Research domain could not have access to the entire University domain (including its child domains such as Life Sciences).

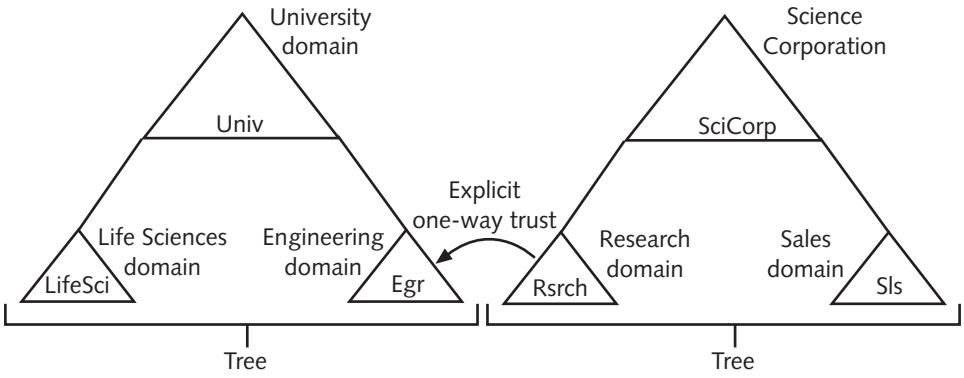

Figure 8-16 Explicit one-way trust between domains in different trees

This section introduced you to the basic concepts of a Windows 2000 network structure. If you are charged with establishing a new Windows 2000 network, you will need to learn a lot more about Active Directory. In that case, you'll want to buy a book on the topic, and perhaps take a class exclusively devoted to Active Directory.

Naming Conventions

In the preceding section you learned to think about domains in terms of their hierarchical relationships. Getting to know the structure of a network by studying its domain tree is similar to understanding your ancestry by studying a genealogical chart. Another way to look at ancestors is to consider their names and their relationship to you. For example, suppose that John Smith is your grandfather. Depending on the speaker and his relationship to this man, one might recognize him as "John Smith," "My paternal grandfather, John Smith," or simply, "John." In the same way, different types of names, depending on where in the domain they are located, may identify objects in a domain.

Naming (or addressing) conventions in Active Directory are based on the conventions used in the Internet. In Internet terminology, the term **namespace** refers to the complete database of hierarchical names used to map IP addresses to their hosts' names. The Internet namespace is not contained on just one computer. Instead, it is divided into

many smaller pieces on computers at different locations on the Internet. In the genealogy analogy, this would be similar to having part of your family records in your home file cabinet, part of them in the state historical archives, part of them in the Ellis Island immigrant files, and part of them in the municipal records of another country. Somewhere in the Internet's vast, decentralized database of names and IP addresses (its namespace), your office workstation's IP address indicates that it can be located at your organization and, further, that it is associated with your computer. In Active Directory, the term namespace refers to a collection of object names and their associated places in the Windows 2000 network. In a genealogy analogy, this would be similar to having one relative (the Active Directory) who knows the names of each family member and how everyone is related. If this relative recorded the information about every relative in a database (for instance, Mary Smith is the wife of John Smith and the mother of Steve and Jessica Smith), this would be similar to what Active Directory does via its namespace.

Because Active Directory namespace follows the conventions of the Internet's namespace, when you connect your Windows 2000 network to the Internet, these two namespaces are compatible. For example, suppose you work for a company called Trinket Makers, and a few years back you contracted with a Web development firm to create a Web site. Further, suppose that the firm chose the Internet domain name "trinketmakers.com" to uniquely identify your company's location on the Internet. When you plan your Windows 2000 network, you will want to call your root domain "trinketmakers" to match its existing Internet domain name (the ".com" part is assumed to be a domain). That way, objects within the Active Directory namespace can be assigned names related to the :trinketmakers.com" domain name, and they will match the object's name in the Internet namespace, should that be necessary.

Each object on a Windows 2000 network can have three different names, as described in the following list:

- **Distinguished name (DN)**—A long form of the object name that explicitly indicates its location within a tree's containers and domains. A distinguished name includes a domain component (DC) name, the names of the domains to which the object belong, an organizational unit (OU) name, the names of the organizational units to which the object belongs, and a common name (CN), or the name of the object. A common name must be unique within a container. In other words, you could have a user called "Msmith" in the Legal container and a user called "Msmith" in the Accounting container, but you could not have two users called "Msmith" in the Legal container. Distinguished names are expressed with the following notation: DC=domain name, OU=organizational unit name, CN=object class, CN=object name. For example, the user Mary Smith in the Legal OU of the trinketmakers domain would have the following distinguished name: DC=Com, DC=trinketmakers, OU=Legal, CN=Msmith. Another way of expressing this distinguished name would be trinketmakers.com/legal/msmith.

- **Relative distinguished name (RDN)**—A name that uniquely identifies an object within a container. For most objects, the relative distinguished name is the same as its common name (CN) in the distinguished name convention. A relative distinguished name is an attribute that belongs to the object. This attribute is assigned to the object when the administrator creates the object (as you will learn to do later in this chapter). Figure 8-17 provides an example of an object, its distinguished name, and its relative distinguished name.

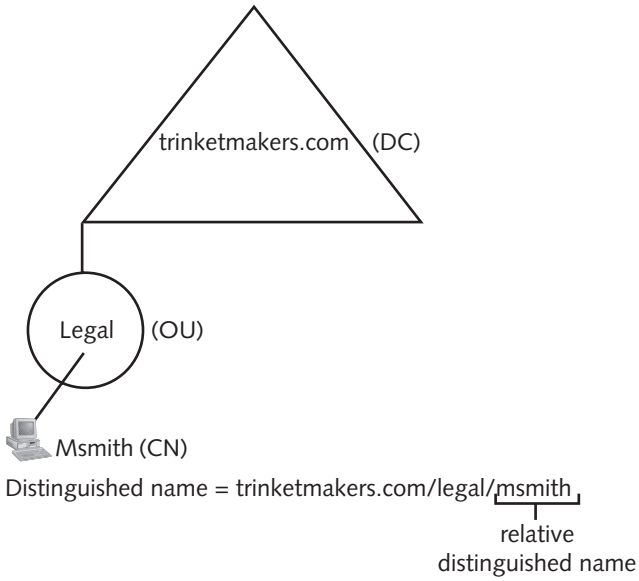

Figure 8-17 Distinguished name and relative distinguished name

- **User principal name (UPN)**—The preferred naming convention for users in e-mail and related Internet services. A user's UPN looks like a familiar Internet address, including the positioning of the domain name after the @ sign. When you create a user account, the user's login name is added to a **UPN suffix**, the portion of the user's UPN that follows the @ sign. A user's default UPN suffix is the domain name of his or her root domain. For example, if Mary Smith's user name is msmith and her root domain is trinketmakers.com, her UPN suffix is trinketmakers.com, and her UPN is *msmith@trinketmakers.com*.

The naming conventions used by Windows 2000 follow those specified in the **Lightweight Directory Access Protocol** (**LDAP**), a protocol for accessing network directories. LDAP relies on TCP/IP and can be used by any modern NOS. Because it is a standard, LDAP allows any application to access the directory of any system according to a single naming convention.

In addition to these names, each object has a **globally unique identifier (GUID)**, a 128-bit number that ensures that no two objects have duplicate names. The GUID is generated and assigned to an object upon its creation. Rather than use any of the alphabetical names, network applications and services communicate with an object via the object's GUID.

Now that you have been introduced to the Windows 2000 Server Active Directory structure and naming conventions, you are ready to learn about installing the NOS.

PLANNING FOR INSTALLATION

When installing and configuring a network operating system, before you insert the installation CD, you must create a plan for your server and its place in your network. You need to consider many factors, including organizational structure, server function, applications, number of users, LAN architecture, and optional services (such as remote access) when developing this plan. Once you have installed and configured the network operating system, changing its configuration may prove difficult and cause service disruptions for users.

The value of planning for installation cannot be overemphasized. Any seasoned network administrator can probably tell a story about a server operating system installation for which he or she should have prepared better. Poor planning results in more work for the installer, potential downtime for users, and headaches for whoever supports the server after installation. To prepare for installation you must first ensure that your server hardware meets the Windows 2000 Server requirements (see Table 8-2). Next, you must prepare answers to the following list of critical preinstallation decisions.

- *How many, how large, and what kind of partitions will the server require?* Windows 2000 must be installed on a single partition. When you install it, you will have a choice of:
  - Creating a new partition on a nonpartitioned portion of a hard disk
  - Creating a new partition on a partitioned hard disk
  - Installing Windows 2000 on an existing partition
  - Removing an existing partition and creating a new one for installation

The option you choose will depend on how your server is currently partitioned, whether you wish to keep data on existing partitions, and how you want to subdivide your server's hard disk. If you know the number and size of the partitions you need (for example, on a 16-GB hard disk you might want to create a 6-GB system partition and a 10-GB data partition), it is best to create them during installation.

- *What type of file system will the server use?* Recall from the discussion about Windows 2000 file systems that the optimal file system for a Windows 2000 server is NTFS. Choose NTFS unless your applications require a different file system. NTFS must be used if you intend to use Active Directory and the domain model for centralized resource and client management.

- *What will the server's name be?* You may use any name that includes a maximum of 15 characters, but you cannot use the following characters: > < [ ] : ; | = , + * " ? Choose a practical, descriptive name that distinguishes the server from others and that is easy for you and your users to remember. For example, you might use geographical server names, such as Boston or Chicago. Alternately, you might name servers according to their function, such as Marketing or Research. If the server is a member of a large domain, you might identify it in relationship to its domain name. For example, the Marketing server in the Pittsburgh domain might be called Pitts-Mktg.
- *Which protocols and network services should the server use?* Before you begin installing Windows 2000 Server, you need to know which protocol (or protocols) your network requires. Recall from Chapter 3 that most organizations are moving toward TCP/IP-based transmission because it is flexible, reliable, and widely supported. On Windows 2000 Server, TCP/IP is the default protocol, and depending on your circumstances, you should probably leave it as such. If your server runs Web services or requires connectivity with UNIX systems, you *must* run TCP/IP. Install NetBEUI on your Windows 2000 server only if you need to communicate with computers running Windows for Workgroups. If your Windows 2000 server must communicate with a NetWare server that does not rely on TCP/IP (for example, a server running NetWare version 3.11), you should also install the NWLink IPX/SPX Compatible Protocol and Gateway Services for NetWare.
- *What will the Administrator password be?* Use a strong password—in other words, one that is difficult to crack. It should consist of at least eight characters, include both letters and numbers, and not resemble any known English words, particularly words that have some association to you or your company. For example, the password "GIANTS" is not secure, while the password "GZ477OPS1" is more secure.
- *Should the network use domains or workgroups, and, if so, what will they be called?* First, you must decide whether your Windows 2000 network will use workgroups or domains. During installation you will be asked whether the server should join an existing workgroup, be a new workgroup server, or join an existing domain. As you learned, in a workgroup situation, computers share network access in a peer-to-peer fashion. It is more likely that your environment will require domains, in which the security for clients and resources is centralized. If the server will be joining an existing domain, you must know the domain name, domain controller name, and the DNS server name. Domain names should describe the logical group of servers and users they support. You may use any name that includes a maximum of 15 alphanumeric characters, but not the following characters into the names: > < [ ] : ; | = , + * " ? Popular schemes for naming domains incorporate geography and function into the names. For example, in a domain model for a WAN spanning several cities, you might want to name your domains Boston,

Chicago, Detroit, Pittsburgh, and so on. In a very large organization, you may want to use a less limiting convention. For example, if your company's business is chemical production, you might want to name your domains Hydrocarbons, Resins, Fertilizers, and so on.

- *Will the server support additional services?* During installation, you will be asked to choose which services your server will support. Of course, you must install certain protocols and network services in order for clients to access the server. You may also want to install optional services, such as: Internet Information Services (applications for creating and hosting Web sites), Terminal Services (to enable remote networking via "thin" clients), Windows Media Services, Message Queuing, and Management and Monitoring Tools. Although it's easiest to include additional services during the original installation, they can be added later as well.
- *Which licensing mode should I choose?* You may choose one of two licensing modes: per seat or per server. The **per server** licensing mode allows a limited number of clients to access the server simultaneously. (The actual number is determined by your Windows 2000 Server purchase.) In per server mode, any of your organization's clients may be capable of connecting to the server. The number of concurrent connections is restricted. Per server mode is a popular choice in organizations that have a limited number of servers and many users, or where multiple users share workstations (for example, a mail-order catalog's call center). The **per seat** mode requires a license for every client capable of connecting to the Windows 2000 server. In environments that include multiple Windows 2000 servers and in which each user has his own workstation, this choice is probably more economical than per server licensing.

If you are running a Windows 2000 server as a Web or FTP server for anonymous clients (for example, Internet users from anywhere in the world), you do not need separate Windows 2000 Server client licenses for these types of clients.

- *How can I remember all of this information?* As you make these preinstallation decisions, you should note your choices on a server installation form and keep the form with you during installation. Appendix D offers an example of such a form.

The preceding list describes only the most significant installation options. You should also be prepared to:

- Read and accept the license agreement
- Identify your organization
- Provide your registration key
- Select the appropriate time and date

- Specify display settings
- Identify and supply drivers for hardware components such as video cards, network adapters, printers, and so on

If you are upgrading from Windows NT Server to Windows 2000 Server, your preparation will include the additional considerations discussed in the following list. (If you are not familiar with Windows NT, this list may include some concepts that are unfamiliar to you.):

- Back up the existing Windows NT server, including its Registry, so that you can return to a working state in case the Windows 2000 Server installation fails.
- On a Windows NT network that includes multiple servers and domain controllers, you should upgrade your Windows NT primary domain controller (PDC) to a Windows 2000 domain controller first. Servers upgraded after the Windows NT PDC has been upgraded can become either domain controllers or member servers.
- Be certain to select the "Upgrade to Windows 2000 (Recommended)" option on the first setup screen.
- If you are upgrading the Windows NT PDC to a Windows 2000 domain controller, indicate that you want to start a new domain or forest during the Active Directory Setup Wizard. On subsequent server upgrades you can choose to join this existing domain.
- After all Windows NT servers on your network are upgraded to Windows 2000 Server, you should convert the domains to native mode. This will indicate to the network that no more Windows NT primary domain controllers (PDCs) or backup domain controllers (BDCs) remain on the network. You can do this by clicking Start/Programs/Administrative Tools/Active Directory Domains and Trusts. Right-click the domain you want to convert and click Properties. Select the General tab and click the Change Mode button.
- Carefully follow the upgrade instructions that come with your Windows 2000 Server software or consult the Microsoft Windows 2000 Server Web site for complete information.

If you are installing Windows 2000 Server on a computer currently running Windows 9x, you must choose the "Clean Install" option, which will overwrite the current operating system. In this instance, you cannot choose the "Upgrade to Windows 2000" option. Therefore, it is very important to create a full backup of your system in case the Windows 2000 Server installation fails and you need to revert to Windows 9x.

Installing and Configuring a Windows 2000 Server

Once you have devised a plan for your Windows 2000 server installation, you can begin the actual installation process. In this section, you will learn about the available options and the decisions you must make when installing and initially configuring your Windows 2000 server.

The Installation Process

When you begin installing Windows 2000 Server, you can install the server from a CD-ROM or remotely over the network. The most popular method of installing Windows 2000 is from a CD-ROM drive. If you must use the network method, be aware that this type of installation generates a high volume of network traffic and shouldn't be performed while clients are attempting to use the network.

The following summary of the Windows 2000 Server installation process assumes, for simplicity, that you are using a CD-ROM. This installation also assumes your computer has a single partition and that you will convert this partition from FAT32 to NTFS. It represents a typical, simple installation. It does not take into consideration any anomalies you might encounter with your server or network environment. As a result, your installation may not proceed effortlessly. The first part of the installation presents GUI interface, then it reverts to a text-based interface, and later a GUI interface once again takes over.

To install Windows 2000 Server from a CD-ROM on a server or workstation already running a Windows 9.x operating system:

1. Make sure there is no disk in the floppy disk drive and then boot the server or workstation.
2. Insert the Windows 2000 Server installation CD into the server's CD-ROM drive. The Microsoft Windows 2000 CD window appears.
3. If your CD-ROM drive is configured to automatically run the program on the CD, you will be asked whether you want to upgrade to Windows 2000. To install Windows 2000 Server, click **Yes**.

   If your CD-ROM drive is not configured to automatically run the program on the CD, the Microsoft Windows 2000 CD window offers you four options: Install Windows 2000, Install Add-on Components, Browse this CD, or Exit. Click the **Install Windows 2000** option to continue.
4. If you are attempting to install Windows 2000 Server on a machine that is already running Windows 9x, a Windows 2000 Setup message appears instructing that you cannot upgrade from this version of the operating system. This means you must perform a clean install (in other words, one that creates an entirely new copy of the operating system on the computer). Click **OK** to continue.

Otherwise, the Windows 2000 Setup window asks whether you want to upgrade to Windows 2000 or install a new copy of Windows 2000. Click the **Install a new copy of Windows 2000 (Clean Install)** option button, and then click **Next** to continue.

5. The Windows 2000 Setup - License Agreement window appears. Read the complete agreement, click the **I accept this agreement** option button, and then click **Next** to continue.
6. The Windows 2000 Setup – Your Product Key window appears. On the Windows 2000 Server package, find your product key, which uniquely identifies your copy of Windows 2000 Server. Enter your product key and click **Next** to continue.
7. The Windows 2000 Setup – Select Special Options window appears. Use this window to choose special options, such as languages and accessibility options, for your Windows 2000 Server install. Click the **Language Options** button.
8. The Language Options dialog box opens, where you can select your default language. Verify that your preferred language (probably English) is selected. If it is, click **OK**. Otherwise, click the list arrow, click your preferred language, and then click **OK**. You return to the Windows 2000 Setup – Select Special Options screen. Click **Next** to continue.
9. The Windows 2000 Setup – Upgrading to the Windows 2000 NTFS File System window appears, urging you to upgrade your drive to the NTFS file system, if it does not already use this file system. Make sure the **Yes, upgrade my drive** option is selected, and then click **Next** to continue.
10. The Microsoft Windows 2000 Server Setup – Directory of Applications for Windows 2000 window appears. You can use this window to connect to Microsoft's Web site to view the Directory of Applications for Windows 2000. This directory lists applications and describes to what extent they have been tested with the Windows 2000 Server operating system. Click **Next** to continue.
11. The Microsoft Windows 2000 Server Setup – Copying Installation Files window appears, and Setup begins loading an information file that contains data you have entered about your installation, and then starts copying installation files to your computer's hard drive. Once those files have been copied, Setup restarts your computer and continues the installation process. You should not need to interfere with this process.
12. After rebooting, the text-based Windows 2000 Server Setup – Welcome to Setup window appears, as shown in Figure 8-18.

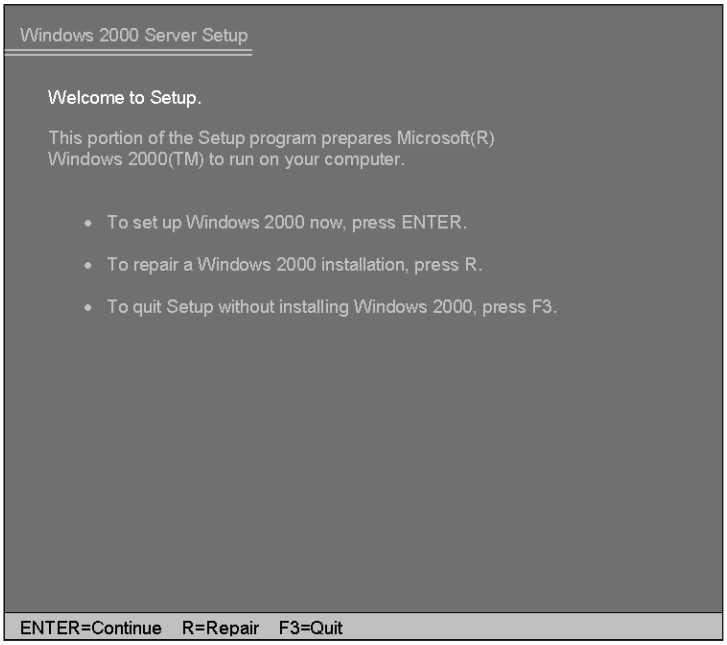

Figure 8-18 Beginning setup options

Press **Enter** to continue with the Windows 2000 Setup process.

13. If you are installing Windows 2000 Server on a computer that already contains a version of this operating system, you can choose to attempt to repair the existing installation or install a new copy of Windows 2000 over the current version. Press **Esc** to install a new copy of the software.
14. Setup identifies the partitions on your computer's hard disk. You can install Windows 2000 Server in an existing partition, create a new partition for this installation, or delete an existing partition. These steps show you how to create an NTFS partition. To begin creating this partition, use the **Up** or **Down Arrow** key to select the partition on which you want to install Windows 2000 Server, and then press **Enter**.
15. If you receive a warning message about installing Windows 2000 on a partition that contains another operating system, continue by pressing **C**.
16. If the partition you selected in Step 14 was not already using the NTFS file system, a warning message informs you that you should not convert the drive to NTFS if you must access the drive via an older operating system. Press **C** to confirm that you want to convert the drive.

17. If you selected an unformatted partition in Step 14, a warning message informs you that existing files on the selected partition will be deleted. Press **F** to format the partition.
18. After the disk is formatted, Setup examines the space on your server's hard disk, and then copies files to the Windows 2000 installation folders. Wait until the process is completed and Setup restarts the computer.
19. After restarting, if the partition you chose in Step 14 was using the FAT32 file system, Setup now converts the partition to NTFS, and then restarts the computer.
20. After restarting, the Windows 2000 Setup program begins again, this time displaying a graphical interface. Eventually, the Windows 2000 Server Setup Wizard screen appears, and then the Windows 2000 Server Setup – Installing Devices window appears. Setup attempts to identify your server's basic hardware and install the device drivers necessary to use that hardware (your screen may momentarily turn black at some point in this process). Although Setup may choose the wrong devices on occasion, it is usually correct. If you have a special NIC, mouse, keyboard, monitor, or other hardware device for which you want to install specific drivers, you can choose to change the components after the program detects them.
21. Once Setup has detected your hardware devices, it prompts you to choose regional settings. Make sure the system locale and keyboard layout are correct, and then click **Next** to continue.
22. In some cases, the Windows 2000 Server Setup – Regional Settings window appears. If you see this window, you can use it to customize the system locale (the identification of your geographic location, which determines the format of numbers, currency, and dates) and your keyboard layout. If you don't see this window, skip to Step 23. Click **Next** to continue.
23. The Windows 2000 Server Setup – Personalize Your Software window appears. To personalize Windows 2000, type your name in the Name text box and then the name of your organization in the Organization text box. Note that these names do not affect your workgroup or domain names for the network.
24. The Windows 2000 Server Setup – Licensing Modes window appears. Setup asks you to select the licensing mode you want to use, either per server or per seat. Choose **per server** licensing if you are planning a network in which only a certain number of clients can be connected to that server at one time. Choose **per seat** licensing if you are guaranteeing that every registered client can connect to the server at any time. Then click **Next** to continue.
25. The Windows 2000 Server Setup – Computer Name and Administrator Password window appears. Enter a computer name and administrator password for your Windows 2000 Server. Use a computer name that reflects the server's purpose. For example, if the server will provide resource sharing to employees in the Marketing Department, you might want to call it "MARKETING-01."

If the server will supply Web pages to the Internet, you might want to call it "WEB_SERVER-01." Standard computer names may contain letters, numbers, and hyphens (-), but no spaces. (If you use a nonstandard computer name, other computers on the network may not be able to easily find the server on the network.) Choose a strong password as described in the installation planning process section. Then click **Next** to continue.

26. The Windows 2000 Server Setup – Windows 2000 Components window appears, where you can add or remove Windows 2000 Server components. (See Figure 8-19.) Scroll down the list of components and click the check boxes next to the following options to add them: **Management and Monitoring Tools**, **Networking Services**, and **Other Network File and Print Services**. Click **Next** to continue.

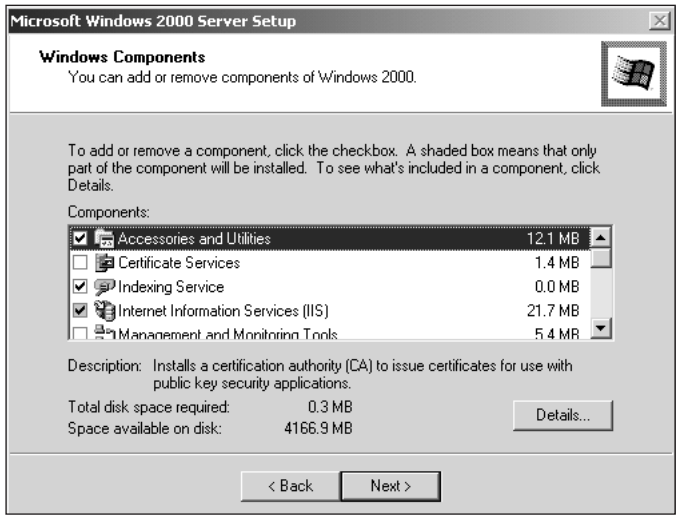

Figure 8-19 Selecting Windows 2000 components

27. If your server has a modem and communication services are installed, the Windows 2000 Server Setup – Modem Dialing Information window appears. Enter modem dialing information by selecting your country and typing your area code. Then click **Next** to continue.

28. The Windows 2000 Server Setup – Date and Time Settings window appears. Select the correct date, time, and time zone, and then **Next** to continue.

29. The Windows 2000 Server Setup – Networking Settings window appears, and Setup installs the networking software that enables your server to connect with other devices on the network. Once this software has been installed, you can choose typical or custom settings for your networking preferences (for example, clients and protocols). Because the typical settings option includes

the most commonly used protocol (TCP/IP), client software (Client for Microsoft Networks), and service (File and Print Sharing for Microsoft Networks), you can accept the default selections. Click **Next** to continue.

30. The Windows 2000 Server Setup – Workgroup or Computer Domain window appears, where you specify whether this server belongs to a domain and, if so, the name of the domain to which it belongs. For the purposes of this installation (considering your server may not be connected to a real network), accept the default option of not belonging to a workgroup or domain and click **Next** to continue.

31. Setup begins installing Windows 2000 components, displaying the Windows 2000 Server Setup – Installing Components window shown in Figure 8-20.

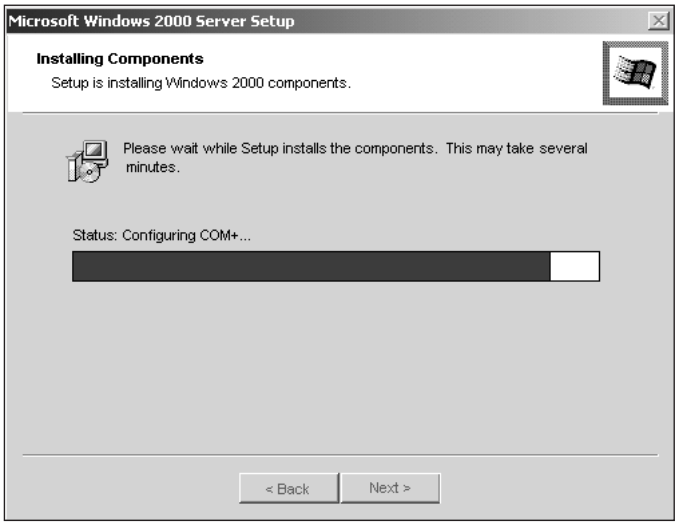

Figure 8-20 Installing Components window

32. Near the end of the process you may see a message window informing you that you should use only statically assigned IP addresses for this server (if it is to be used as a WINS server), and that you will have the option to change the existing, dynamically assigned IP address. Click **OK** to continue. If you do not see this message window, you can skip to Step 36.

33. The Local Area Connection Properties dialog box appears. Click the **Internet Protocol** component to highlight it, and then click **Properties**.

34. The Internet Protocol (TCP/IP) Properties dialog box appears. Click the **Use the following IP address** option button to begin assigning the server a static IP address. For this installation (assuming you are not going to use this server to connect to external LANs or WANs), enter the IP address **100.100.100.100** and the default gateway **100.100.100.1**. The subnet mask is automatically assigned a default value of 255.0.0.0. Click **OK** to save your changes.

35. You return to the Local Area Connection Properties dialog box. Click **OK** again to save your changes and continue.
36. The Windows 2000 Server Setup – Completing the Windows 2000 Setup Wizard appears. Click **Finish** to complete the setup process.

Your computer restarts, signaling that the Windows 2000 Server installation is complete. A Welcome to Windows window appears. In the next section you will learn how to log on for the first time, and then configure your newly installed NOS.

This sample installation uses the default selections and simplest methods of configuring the server. In reality, your server installations may not be as straightforward.

Initial Configuration

Although you have completed the Windows 2000 Server installation, the server isn't yet ready to support clients on a network. First you must configure the software (for instance, assign it a place in the domain).

To configure Windows 2000 Server:

1. After installing Windows 2000 Server, press **Ctrl+Alt+Del** when prompted, to access a login screen. Enter the default user name **Administrator** (if it has not been entered for you), and the password you chose in Step 25 of the preceding section.

To enhance security, at some point you should create a new user ID with administrative privileges to perform network administration, and disable the Administrator user ID. If you keep the Administrator ID active with full privileges, hackers have half the information they need to break into your system.

2. The Windows 2000 Configure Your Server wizard starts, preparing to guide you through configuring the server. First, you specify whether this server is the only server on the network or is one of many servers on the network. For the purposes of this exercise, click the **This is the only server in my network** option button, and then click **Next**.
3. The next window in the wizard informs you that Windows will configure the server as a domain controller and set up Active Directory, DNS, and DHCP on your network. For the purposes of this exercise, click **Next** to accept these conditions.
4. In the next window in the wizard, enter a domain name to create a domain. Type the name of the domain that this server will control. For example, if this server will control all resources for the executives in the organization, you might want to call it "EXECUTIVES."

5. Type **local** in the registered domain name text box. This option assumes that your server will not connect to the Internet. Click **Next** to continue.
6. You see a message indicating that the computer must be restarted so your settings can take effect. Click **Next** to restart your server.
7. The c:\WINNT\System32\netsh.exe window appears, and the Windows Components Wizard begins configuring components according to the information you specified. Then the Configuring Active Directory window appears, indicating that the wizard is configuring Active Directory and that the process may take a while.
8. Eventually, your computer restarts and Windows 2000 Server presents the Log On to Windows window. In the next section, you will learn how to configure users, groups, and other objects on your Windows 2000 Server.

Establishing Users, Groups, and Rights

Earlier in this chapter you learned how networks manipulate user and group accounts to restrict or allow access to specific resources. Now you are ready to learn how to establish users and groups through the Windows 2000 Server interface.

After installation, your Windows 2000 Server will already have two predefined accounts: Guest and Administrator. The **Guest** account is a predefined user account with limited privileges that allows a user to log onto the computer. The **Administrator** account is a predefined user account that has the most extensive privileges for resources both on the computer and on the domain that it controls (if it is a domain controller). These two predefined user accounts are designed primarily to allow you to log onto a computer after installation and before you have created any additional user accounts. The Guest and administrator accounts cannot be deleted; however, they may be disabled. Additional accounts that you create may be **local accounts**, or those that only have rights on the server they are logged onto and **domain accounts**, those that have rights throughout the domain. To create domain accounts, you must have Active Directory installed and your domains properly configured. This exercise assumes that Active Directory is installed on your Windows 2000 server and that domains are in use.

To create a domain user account:

1. Make sure you are logged on as Administrator.
2. Click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Active Directory Users and Computers**. The Active Directory Users and Computers snap-in opens.
3. Double-click the Active Directory container in which you want to create the new user. This may be a domain or an OU.
4. Right-click the **Users** folder, point to **New** on the shortcut menu, and then click **User**. The New Object – User dialog box appears.

5. Type the user's last and first name in the appropriate text boxes. You then see the user's full name in the Full name text box.
6. Enter a user name in the User logon name text box. This name uniquely identifies the user in a domain or forest. The domain name is provided automatically. Click **Next** to continue.
7. In the New Object – User dialog box, enter a password for the user, as shown in Figure 8-21. Enter a strong password (one that consists of at least eight characters, cannot be found in the dictionary, and contains both numbers and letters). Retype the password in the Confirm password text box. You may also select from four additional options: User must change password at next logon, User cannot change password, Password never expires, or Account is disabled. It's a good policy to force the user to pick a new password the first time they log in, so that they have a password that is meaningful to them and so that you, as the network administrator, don't know their password. It is also a good policy to allow the password to periodically expire. With this in mind, check the first option, **User must change password at next logon**, and then click **Next**.

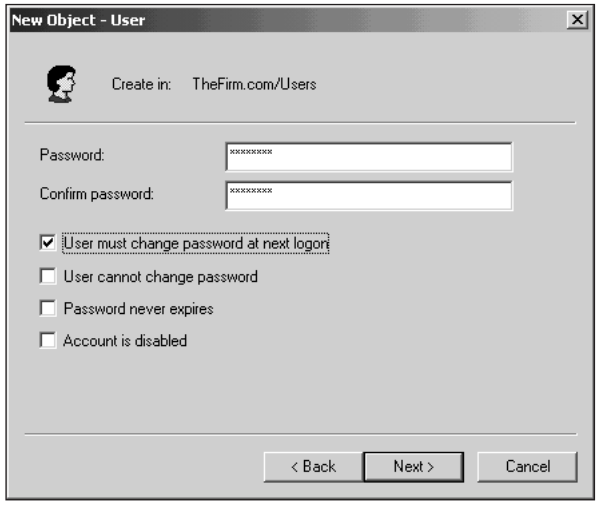

Figure 8-21 User account password properties

8. The next New Object – User window displays the information you have entered. Click **Finish** to complete the creation of a new domain user account.

Once you have created a new user, configure the properties for that user, including their address, telephone number, and e-mail address, their rights to use remote access, their position in the organization, their group memberships, what hours of the day they may log onto the network, and so on. To modify user account properties, you can use the Active Directory Users and Computers snap-in. In the snap-in window, double-click the

user account in the right-hand pane. The User Account Properties dialog box opens, with multiple tabs that represent different categories of attributes you may change.

Before you add many users, you will probably want to establish groups into which you can collect user accounts. But before creating a group, you must know what type of scope the group will have: domain local, global, or universal. The group's scope identifies how broadly across the Windows 2000 network its privileges can reach. A **domain local group** is one that allows its members access to resources within a single domain. Domain local groups are used to control access to certain folders, directories, or other resources. They may also contain global groups. A **global group** allows its members access to resources within a single domain also. However, a global group usually contains user accounts and can be inserted (or nested) into a domain local group to gain access to resources in other domains. A **universal group** is one that allows its members access to resources across multiple domains and forests.

To create a group in Windows 2000 Server:

1. First make sure you are logged on as Administrator.
2. Click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Active Directory Users and Computers**. The Active Directory Users and Computers snap-in starts.
3. Double-click the Active Directory container in which you want to create the new user. This may be a domain or an OU.
4. Click the **New Group** button (which looks like two faces in profile) on the toolbar. The New Object – Group dialog box appears, as shown in Figure 8-22.

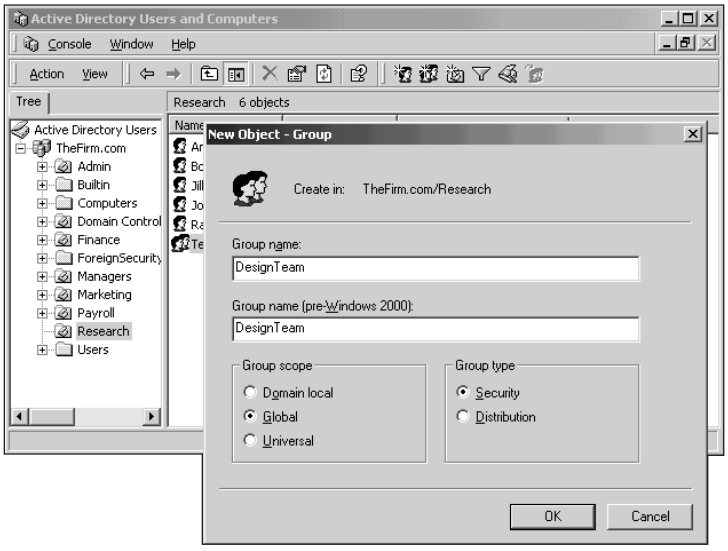

Figure 8-22 Creating a group

5. In the New Object – Group dialog box, enter the name of the group in the Group name text box. In case you are using both Windows 2000 and Windows NT servers on your network, the Group name (pre-Windows 2000) text box is automatically completed.
6. Choose the group scope: Domain local, Global, or Universal. Select the type of group you want to create.
7. Select the type of group you want to create: Security or Distribution. A security group can be assigned access to resources, while distribution groups are used solely for e-mail distribution. Once you have made your selection, click **OK** to complete creating the new group.

Modifying the properties of a group account is similar to modifying the properties of a user account. To do so, double-click the group in the right pane of the Active Directory Users and Computers snap-in window. This opens the group's Properties dialog box, which contains four tabs: General, Members, Member Of, and Managed By. Through this dialog box you can add user accounts to the group, make the group a member of another group, and identify a user account to manage the group.

As mentioned earlier, users and groups are virtually useless unless they have some rights to the server's data and system directories. As an example, the following steps describe how to assign a group called "Instructors" permission to access and modify data in the server's Program Files directory. These steps assume that your server's disk uses the NTFS file system.

To modify the permissions for a directory:

1. Double-click the **My Computer** icon on the Windows 2000 Server desktop. The My Computer window opens.
2. Double-click the **Local Disk (C:)** icon. The Local Disk (C: drive) window opens.
3. Right-click the **Program Files** folder, and then click **Properties** on the shortcut menu. The Program Files Properties dialog box opens, with four tabs: General, Web Sharing, Sharing, and Security, as shown in Figure 8-23.

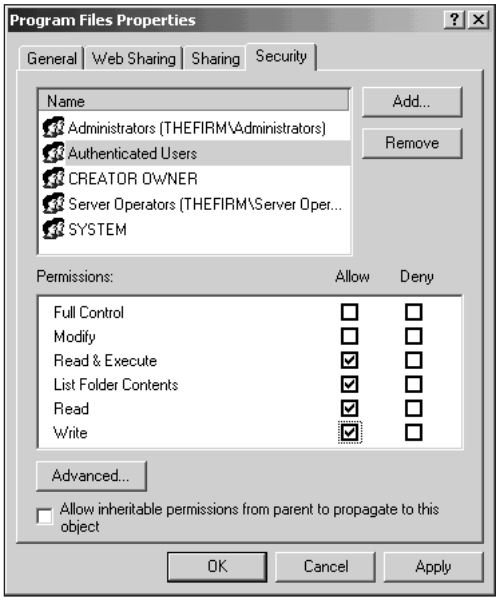

Figure 8-23 The Program Files Properties dialog box

4. Click the **Security** tab.
5. To add a group, click **Add**. The Select Users, Computers, or Groups dialog box opens.
6. Select the **Users** group from the list of users, computers, and groups. Click **Add**, and then click **OK**. The Select Users, Computers, or Groups dialog box closes and you are returned to the Program Files Properties dialog box.
7. Now you need to select the permissions you want to grant to the group. The permissions options are Full Control (allows users to read, add, delete, execute, and modify files and subfolders, and decide who else has permissions in the folder), Modify (allows users to read, add, delete, execute, and modify files), Read & Execute (allows users to view and execute files but not change them in any way), List Folder Contents (allows users to view the directory and subdirectory listings, but not their files), Read (allows users to view files but not execute them or change them in any way), and Write (allows users to create new files, add to existing files, delete files, modify files, and create folders within the folder). In this example, click the check box under "Allow" for the **Modify** permission, and then click **OK**.
8. The group Properties dialog box closes and you return to the Local Disk (C:) window.

Internetworking with Other Network Operating Systems

Windows 2000 Server can communicate with almost any kind of client and, given the proper software and configuration, with the other major NOSs. Interoperability is a major concern, as more organizations face the challenge of dealing with mixed networks. In the interest of the consumer, Microsoft and other network operating system vendors have made efforts to close the gap. You will encounter situations in which Windows 2000 must coexist on the same network with NetWare or UNIX or both. This section focuses on Microsoft's solution to the interoperability question. The next chapter discusses the Novell approach.

You might think that establishing communications between two network operating systems is simply a matter of installing the same protocol on both systems. For example, you might think that because both NetWare and Windows 2000 can run versions of the IPX/SPX protocol, the two should be able to communicate directly. (Recall from Chapter 3 that the NWLink IPX/SPX-compatible is required by NetWare versions 3.x and lower and supported by higher versions of NetWare.) In fact, a protocol match is merely one part of the interoperability equation.

To be fully compatible—in other words, to integrate both print and file services, directories, accounts, and other objects—operating systems must also run compatible redirectors. Windows 2000 and NetWare use different client redirector languages that are incompatible. To bridge this gap on a Windows 2000 server, you have two options: either the Windows 2000 Server can run Microsoft's **Gateway Services for NetWare (GSNW)** or the clients that depend on the Windows 2000 Server can run Microsoft's **Client Services for NetWare (CSNW)**. GSNW is a service that runs on the Windows 2000 Server and acts as a translator, or gateway, between its redirector services and those on the NetWare server. With GSNW and NWLink installed, a Windows 2000 server and its clients can access files and other shared resources on any NetWare server on the network through the NetWare Directory Service (NDS). CSNW is a service that runs on a Windows 2000 client and in conjunction with NWLink enables the client to log on directly to the NetWare server to access its printers, files, and other resources. The advantage to using the gateway service over the client service is that the former requires only one setup. The client services require separate setup for each client. However, the gateway service cannot apply different security levels to each different user and, therefore, may be less secure.

Keep in mind, however, that installing the NWLink IPX/SPX-compatible protocol and GSNW or CSNW may not suffice to allow your two kinds of servers to communicate. You must be careful to configure the NWLink parameters exactly right. If the two network operating systems still do not communicate, you may need to reconfigure NWLink, paying special attention to the Frame Type option.

If you are in an environment that contains both Windows 2000 and NetWare servers, and both use TCP/IP as their preferred protocol, you do not necessarily have to install

GSNW or CSNW (nor do you need to install NWLink). Instead, on each workstation you could install Novell's recommended client software (as described in the next chapter) to access NetWare servers in addition to Microsoft's Client for Networks to access Windows 2000 servers. If for some reason you did not want to install Novell's client on your workstations, you could use the Windows 2000 Server with GSNW installed (and use the TCP/IP protocol rather than the NWLink protocol) as a means for those clients to access NetWare resources.

Even though you may enable clients to access a NetWare server through either gateway or client services for NetWare, you still have to create the user accounts and provide them permissions in the NetWare NOS. You will learn how to do this in the next chapter.

Windows 2000 Server also comes with a migration utility, called the **Directory Services Migration Tool (DSMIGRATE)** that enables you to migrate accounts, groups, files, and permissions from a NetWare NDS directory to the Windows 2000 Server Active Directory.

Interconnecting UNIX and Windows 2000 networks is somewhat easier, because you can assume that both rely on the TCP/IP protocol. In order for clients on Windows 2000 networks to access UNIX servers and make use of their files and account privileges, you would install Microsoft's Services for UNIX on each client. The Services for UNIX include the ability for the client to be recognized by UNIX's file system and utilities for manipulating UNIX files and directories. You will learn more about UNIX client connections in Chapter 10.

CHAPTER SUMMARY

- Network operating systems are entirely software-based and can run on a number of different hardware platforms and network topologies.
- Network administrators choose an appropriate NOS according to what's compatible with the existing infrastructure; whether it supports the applications, services, and security required by the environment; whether it will grow with the organization; whether the vendor will provide reliable technical support; and whether it fits in the budget.
- A redirector, which belongs to the Presentation layer of the OSI Model, is inherent in both the network operating system and the client operating system. On the client side, it intercepts client communications and decides whether the request is meant for the server or for the client.
- When a client attempts to log on, the network operating system receives the client's request for service and tries to match the user name and password with the name and password in its user database. If the passwords match, the NOS grants the client access to resources on the network, according to limitations. This process is known as authentication.
- Users with similar needs and restrictions are collected in groups to more easily manage their access and privileges.

- A directory is an NOS's method of organizing and managing objects, such as users, printers, server volumes, and applications. It is sometimes compared to a tree, because it has one common starting point and branches into multiple containers, which may branch into additional containers.
- A file system is an operating system's method of organizing, managing, and accessing its files through logical structures and software routines. In general, when installing a Windows 2000 server, you will want to choose the NTFS file system.
- In order for clients to share a server application, the network administrator must assign users rights to the directories where the application's files are installed. Users will at least need rights to access and read files in those directories. For some applications, you may also need to give users rights to create, erase, or modify files associated with the application.
- In order for clients to share a network printer, the printer must be created as an object, assigned a name and properties, and then shared among clients. Users or groups may be assigned different levels of privileges to operate printers.
- The type of multitasking supported by NetWare, UNIX, and Windows 2000 Server performs one task at a time, allowing one program to use the processor for a certain period of time, and then suspending that program to allow another program to use the processor. This is called preemptive multitasking.
- Multiprocessing splits tasks among multiple processors to expedite the completion of any single instruction. It's a great advantage for servers with high CPU utilization, because it improves performance. Windows 2000 Server and Netware support symmetric multiprocessing, which splits all operations equally among two or more processors.
- Windows 2000 supports any type of topology or protocol you are likely to run on a LAN. This efficient network operating system uses multiple processors and employs multitasking to allow processes on the server to share CPU resources. It's also easy to manage and well supported.
- Windows 2000 Server requires the following minimum hardware: Pentium processor with a minimum clock speed of 133 MHz, 128 MB RAM, at least 1 GB free hard disk space for system files (but 2 GB is recommended), and a pointing device. A CD-ROM and a NIC that are included on Microsoft's Hardware Compatibility List (HCL) are optional. By default, it supports a maximum of four processors on one server.
- Windows 2000 Server's memory model assigns each process its own 32-bit memory area. This memory area is a logical subdivision of the entire amount of memory available to the server. Assigning processes separate areas makes the processes less prone to interfering with each other when they run simultaneously.
- The description of object types, or classes, and their required and optional attributes that are stored in Active Directory is known as a schema.
- Domains define a group of systems and resources that share common security and management policies. The database that domains use to record their objects and

attributes is contained within Active Directory. Domains are established on a network to make it easier to organize and manage resources and security.

- When multiple domain controllers are used, a change to the database contained on one domain controller is copied to the databases on other domain controllers so that their databases are always identical. The process of copying directory data to multiple domain controllers is known as replication.
- To collect domains into logical groups, Windows 2000 Server uses a domain tree (or simply, tree). At the base of the tree is the root domain. From the root domain, child domains branch out to separate objects with the same policies. Underneath the child domains, multiple organizational units branch out to further logically subdivide the network's systems and objects. A collection of domain trees is known as a forest.
- Each tree, domain, container, and object has a unique name that becomes part of the namespace. The names of these elements may be used in one of three different ways to uniquely identify an object in a Windows 2000 tree: as a distinguished name, as a relative distinguished name, and as a user principal name.
- Prior to installation, you need to make a number of decisions regarding your server and network pertaining to the domain characteristics, the file system, the disk partitioning, the optional services to be installed, the administrator password, the protocols to be installed, and the server's name.
- If you are integrating Windows 2000 network with a NetWare network running IPX/SPX, you need to install the NWLink protocol on both clients and servers and either Gateway Services for NetWare (GSNW) on the Windows 2000 server or Client Services for NetWare (CSNW) on its clients.

KEY TERMS

3-tier architecture — A client/server environment that uses middleware to translate requests between the client and server.

account — A record of a user that contains all of his or her properties, including rights to resources, password, username, and so on.

Active Directory — Windows 2000 Server's method for organizing and managing objects associated with the network.

Administrator — A user account that has unlimited privileges to resources and objects managed by a server or domain. The administrator account is created during NOS installation.

asymmetric multiprocessing — A multiprocessing method that assigns each subtask to a specific processor.

attribute — A variable property associated with a network object. For example, a restriction on the time of day a user can log on is an attribute associated with that user object.

authentication — The process whereby a network operating system verifies that a client's user name and password are valid and allows the client to log onto the network.

CD-ROM File System (CDFS) — The read-only file system used to access resources on a CD. Windows 2000 supports this file system to allow CD-ROM file sharing.

child domain — A domain found beneath another domain in a Windows 2000 domain tree.

class — A type of object recognized by an NOS directory and defined in an NOS schema. Printers and users are examples of object classes.

Client Services for NetWare (CSNW) — A Microsoft program that can be installed on Windows 2000 clients to enable them to access NetWare servers and make full use of the NetWare Directory System (NDS), its objects, files, directories, and permissions.

clustering — A method for connecting multiple servers to enable resource sharing and load balancing between them.

container — A logical receptacle for holding like objects in an NOS directory. Containers form the branches of the directory tree.

directory — In general, a listing that organizes resources and correlates them with their properties. In the context of network operating systems, a method for organizing and managing objects.

Directory Services Migration Tool (DSMIGRATE) — A tool provided with Windows 2000 Server that enables network administrators to migrate accounts, files, and permissions from a NetWare NDS directory to theWindows 2000 Active Server Directory.

distinguished name (DN) — A long form of an object's name in Active Directory that explicitly indicates the object name, plus the names of its containers and domains. A distinguished name includes a domain component (DC), organizational unit (OU), and common name (CN). A client uses the distinguished name to access a particular object, such as a printer.

domain — A group of users, servers, and other resources that share account and security policies through a Windows 2000 network operating system.

domain account — A type of user account on a Windows 2000 network that has privileges to resources across the domain onto which it is logged.

domain controller — A Windows 2000 server that contains a replica of the Active Directory database.

domain local group — A group on a Windows 2000 network that allows members of one domain to access resources within that domain only.

domain tree — A group of hierarchically arranged domains that share a common namespace in the Windows 2000 Active Directory.

explicit one-way trust — A type of trust relationship in which two domains that belong to different NOS directory trees are configured to trust each other.

extended attributes — Attributes beyond the basic Read, Write, System Hidden, and Archive attrevutes supported by FAT.HPFS supports extended attributes.

FAT16 (16-bit File Allocation Table) — A file system designed for use with early DOS- and Windows-based computers that allocates file system space in 16-bit units. Compared to FAT32, FAT16 is less desirable because of its partition size, file naming, fragmentation, speed, and security limitations.

File Allocation Table (FAT) — The original PC file system designed in the 1970s to support floppy disks and, later, hard disks. FAT is inadequate for most server operating systems because of its partition size limitations, naming limitations, and fragmentation and speed issues.

FAT (File Allocation Table) — An enhanced version of FAT that accommodates the use of long filenames and smaller allocation units on a disk. FAT32 makes more efficient use of disk space than the original FAT.

file system — An operating system's method of organizing, managing, and accessing its files through logical structures and software routines.

forest — In the context of Windows 2000 Server, a collection of domain trees that use different namespaces. A forest allows for trust relationships to be established between trees.

Gateway Services for NetWare (GSNW) — A Windows 2000 service that acts as a translator between the Windows 2000 and NetWare client redirector services. With GSNW installed, a Windows 2000 server can access files and other shared resources on any NetWare server on a network.

global group — A group on a Windows 2000 network that allows members of one domain to access resources within that domain as well as resources from other domains in the same forest.

globally unique identifier (GUID) — A 128-bit number generated and assigned to an object upon its creation in the Windows 2000 Active Directory. Network applications and services use an object's GUID to communicate with it.

graphical user interface (GUI) — A pictorial representation of computer functions and elements that, in the case of network operating systems, enables administrators to more easily manage files, users, groups, security, printers, and other issues.

group — A means of collectively managing users' permissions and restrictions applied to shared resources. Groups form the basis for resource and account management for every type of network operating system, not just Windows 2000 Server. Many network administrators create groups according to department or, even more specifically, according to job function within a department.

Guest — A user account with very limited privileges that is created during the installation of a network operating system.

Hardware Compatibility List (HCL) — A list of computer components proven to be compatible with Windows 2000 Server. The HCL appears on the same CD as your Windows 2000 Server software and on Microsoft's Web site.

High-Performance File System (HPFS) — A file system designed for IBM's OS/2 operating system that offers greater efficiency and reliability than does FAT. HPFS is rarely used but can be supported by Windows 2000 servers.

Lightweight Directory Access Protocol (LDAP) — A standard protocol for accessing network directories.

local account — A type of user account on a Windows 2000 network that has rights to the resources managed by the server the user has logged onto.

member server — A type of server on a Windows 2000 network that does not hold directory information and therefore cannot authenticate users.

Microsoft Management Console (MMC) — A graphical network management interface used with Windows 2000 Server.

middleware — Software that sits between the client and server in a 3-tier architecture. Middleware may be used as a messaging service between clients and servers, as a universal query language for databases, or as means of coordinating processes between multiple servers that need to work together in servicing clients.

multi-master replication — The technique of replicating an Active Directory database to multiple domain controllers so they each have the same data and the same privileges to modify that data. Multi-master replication is used within a domain tree.

multiprocessing — The technique of splitting tasks among multiple processors to expedite the completion of any single instruction.

multitasking — The ability of a processor to perform multiple activities in a brief period of time (often seeming simultaneous to the user).

namespace — The complete database of hierarchical names (including host and domain names) used to resolve IP addresses with their hosts.

New Technology File System (NTFS) — A file system developed by Microsoft for use with its Windows NT and Windows 2000 operating systems. NTFS integrates reliability, compression, the ability to handle massive files, system security, and fast access. Most Windows 2000 Server partitions employ either FAT32 or NTFS.

NWConv — A utility provided with Windows 2000 that converts (migrates) an existing NetWare server's user account, file, and other information to a Windows 2000 server.

object — A representation of a thing or person associated with the network that belongs in the NOS directory. Objects include users, printers, groups, computers, data files, and applications.

object class — See *Class*.

organizational unit (OU) — A container within an NOS directory used to group objects with similar characteristics or privileges.

page file — A file on the hard disk that is used for virtual memory.

paging — The process of moving blocks of information, called pages, between RAM and into a page file on disk.

per seat — A Windows 2000 Server licensing mode that requires a license for every client capable of connecting to the Windows 2000 server.

per server — A Windows 2000 Server licensing mode that allows a limited number of clients to access the server simultaneously. (The number is determined by your Windows 2000 Server purchase agreement.) The restriction applies to the number of concurrent connections, rather than specific clients. Per server mode is the most popular choice for installing Windows 2000 Server.

physical memory — The RAM chips installed on the computer's system board that provide dedicated memory to that computer.

preemptive multitasking — The type of multitasking supported by NetWare, UNIX, and Windows 2000 Server that actually performs one task at a time, allowing one program to use the processor for a certain period of time, then suspending that program to allow another program to use the processor.

printer queue — A logical representation of a networked printer's functionality. To use a printer, clients must have access to the printer queue.

process — A routine of sequential instructions that runs until it has achieved its goal. For example, a spreadsheet program is a process.

redirector — A service that runs on a client workstation and determines whether the client's request should be handled by the client or the server.

relative distinguished name (RDN) — An attribute of the object that identifies an object separately from its related container(s) and domain. For most objects, the relative distinguished name is the same as its common name (CN) in the distinguished name convention.

replication — The process of copying Active Directory data to multiple domain controllers. This ensures redundancy so that in case one of the domain controllers fails, clients can still log onto the network, be authenticated, and access resources.

root domain — In Windows 2000 networking, the single domain from which child domains branch out in a domain tree.

schema — The description of object types, or classes, and their required and optional attributes that are stored in an NOS's directory.

site license — A type of software license that, for a fixed price, allows any number of users in one location to legally access an application.

snap-in — An administrative tool, such as Computer Management, that can be added to the Microsoft Management Console (MMC).

swap file — See *Page file*.

symmetric multiprocessing — A method of multiprocessing that splits all operations equally among two or more processors. Windows 2000 Server supports this type of multiprocessing.

thin client — A type of software that enables a client to accomplish functions over a network while utilizing little of the client workstation's resources and, instead, relying on the server to carry the processing burden.

thread — A well-defined, self-contained subset of a process. Using threads within a process enables a program to efficiently perform related, multiple, simultaneous activities. Threads are also used to enable processes to use multiple processors on SMP systems.

tree — A logical representation of multiple, hierarchical levels in a directory. It is called a tree because the whole structure shares a common starting point (the root) and from that point extends branches (or containers), which may extend additional branches, and so on.

trust relationship — The relationship between two domains on a Windows 2000 or Windows NT network that allows a domain controller from one domain to authenticate users from the other domain.

two-way transitive trust — The security relationship between domains in the same domain tree in which one domain grants every other domain in the tree access to its resources and, in turn, that domain can access other domains' resources. When a new domain is added to a tree, it immediately shares a two-way trust with the other domains in the tree.

Universal Disk Format (UDF) — A file system used on CD-ROMs and digital video disc (DVD) media.

universal group — A group on a Windows 2000 network that allows members from one domain to access resources in multiple domains and forests.

user principal name (UPN) — The preferred Active Directory naming convention for objects when used in informal situations. This name looks like a familiar Internet address, including the positioning of the domain name after the @ sign. UPNs are typically used for e-mail and related Internet services.

user principal name (UPN) suffix — The portion of a universal principal name (in Windows 2000 Active Directory's naming conventions) that follows the @ sign.

virtual memory — Memory that is logically carved out of space on the hard disk and added to physical memory (RAM).

wizard — A simple graphical program that assists the user in performing complex tasks, such as configuring a NIC on a server.

workgroup — A group of interconnected computers that share each others' resources without relying on a central file server.

Review Questions

1. List four factors that you should consider before purchasing a network operating system.
2. What is the function of a redirector?
   a. to route CPU requests to the appropriate IRQ on the client
   b. to enable multiple processes to be handled by the same CPU on the server
   c. to determine whether a request is meant for the client CPU or the server
   d. to balance the processing load between the client and the server
   e. to store unfinished processes in a cache until the server can accept them

3. Which of the following must be installed on a Windows 2000 Professional client workstation in order for it to be able to log onto a Windows 2000 server?
   a. Client Services for Windows 2000
   b. Client Gateway to Windows 2000
   c. Services for TCP/IP networks
   d. Client for Microsoft Networks
   e. Windows-compatible client services
4. What are the 3 tiers in a 3-tier architecture?
   a. client, server, network
   b. client, hub, server
   c. client, middleware, server
   d. client, server, client agent
   e. client, server, router
5. If a user has Modify rights to a folder on a Windows 2000 server, what is he or she able to do?
   a. List, read, add, delete, execute, and modify files in the folder
   b. Add, delete, and modify files in the folder
   c. List, read, add, and delete files in the folder
   d. Add, delete, execute, and modify files in the folder
   e. List, read, add, delete, execute, and modify files in the folder, plus set permissions for the folder
6. Groups can contain other groups. True or False?
7. You have created a printer object for a new HP LaserJet in your Windows 2000 server Active Directory. Before users can print to this printer, what else must you create in Active Directory?
   a. a print queue (or share)
   b. a printer folder
   c. a Printers group
   d. a printer administrator
   e. a logical printer port
8. Name at least five attributes that may be associated with a user account.
9. What is the purpose of a container in an NOS directory?
   a. to represent a person or device on the network
   b. to limit the amount of hard disk space each user can use for data files
   c. to indicate in which domain objects belong

d. to organize similar objects for easier management

e. to separate partitions with different file systems

10. What is the maximum amount of memory that a Windows 2000 server can utilize?

a. 2 GB

b. 4 GB

c. 8 GB

d. 10 GB

e. 12 GB

11. What primary advantage does Windows 2000 gain by assigning each operation its own 32-bit address space?

12. What is the common name in the following distinguished name: widgets.com/charleston/marketing/jkessel?

a. widgets.com

b. com

c. charleston

d. marketing

e. jkessel

13. What do threads have to do with multiprocessing?

a. Threads are made of processes; as processes are split among multiple processors, threads keep track of how they were separated in order for them to be rejoined.

b. Threads are made of processes; in order for a multithreaded application to perform instructions faster, each process should use a separate processor.

c. Processes are made of threads; threads within a process can be handled by different processors to improve server performance.

d. Processes are made of threads; threads within each separate process must use the same processor, but different processes can use different processors.

14. In comparing an NOS directory to a tree, what are analogous to leaves?

a. properties

b. containers

c. OUs

d. objects

e. branches

15. In case RAM runs out of space, where can it store unused information blocks?
    a. ROM
    b. a swap file on the hard disk
    c. a cache on the client workstation
    d. EEPROM
    e. BIOS
16. What kind of trust relationship do multiple domains within the same domain tree on a Windows 2000 network use?
    a. master domain trust
    b. one-way trust
    c. two-way trust
    d. global trust
    e. multi-master domain trust
17. On a Windows 2000 server, two domains, named Marketing and Engineering, are within the same domain tree. Thus, a user from the Engineering domain has access to run any of the programs on a server in the Marketing domain. True or False?
18. When talking about a Windows 2000 domain tree, what would you call the one domain from which all other domains and their containers emanate?
    a. branch domain
    b. child domain
    c. leaf domain
    d. root domain
    e. grandfather domain
19. You are a user on a Windows 2000 network who wants to print a memo to a printer across the hall. What kind of name would your client software use to direct your memo to this printer?
    a. distinguished name
    b. attribute name
    c. relative distinguished name
    d. user principal name
    e. global name
20. Which file system must a Windows 2000 server use to accept Macintosh clients?
    a. FAT
    b. FAT32
    c. CDFS
    d. NTFS
    e. MFS

21. What kind of multiprocessing does Windows 2000 support?
    a. symmetric
    b. synchronous
    c. preemptive
    d. asymmetric
    e. asynchronous
22. Name six things you need to know before you begin installing the Windows 2000 Server operating system.
23. After you install Windows 2000 Server, you want to use the Microsoft Management Console (MMC) to manage users and groups. What is the first thing you must do?
    a. add the Local Users and Groups snap-in to the MMC
    b. run the MMC command from the Run dialog box to initiate a new MMC
    c. install the MMC as an optional service from the Windows 2000 Server CD-ROM
    d. choose MMC from the Control Panel, then modify its properties to assign your console a name
    e. convert the Active Directory schema into an MMC-compliant format
24. What is the purpose of replication on a Windows 2000 network?
    a. to create multiple instances of the Active Directory database in case one server fails
    b. to create multiple instances of groups to enable access across all the domains in a domain tree
    c. to copy one domain tree's Active Directory database to another domain tree
    d. to copy all the user objects from one domain within a tree to another domain within a tree
    e. to create a system backup for the domain controller so that in case it fails, all of its files can be restored
25. What is the maximum length for an NTFS file name?
    a. 8 characters
    b. 16 characters
    c. 64 characters
    d. 128 characters
    e. 255 characters
26. You work at an organization with 12 servers and 1000 employees, each of whom works full-time and has his or her own workstation. Which Windows 2000 licensing mode should you purchase?

27. You are the network administrator for a school district. During August, you create a number of new student user accounts, but you do not want them to be functional at that time. What option should you choose when you create the user accounts?
    a. User must change password at next login
    b. User cannot change password
    c. User requires no password
    d. Password never expires
    e. Account is disabled
28. What utility allows you to transform NetWare directory objects to a Windows 2000 Server's Active Directory?
    a. NWConv
    b. NW-Migrate
    c. DSMIGRATE
    d. OSTranslate
    e. Win2KConv
29. Which of the following is not a valid name for a Windows 2000 domain?
    a. MKTG+SALES
    b. MKTG_SALES
    c. MKTGandSALES
    d. MKTG-SALES
    e. MarketingandSales
30. What two methods can you use for installing Windows 2000 Server?
    a. remote installation over the network
    b. installation from multiple floppy disks
    c. installation from a locally attached DAT tape drive
    d. installation from a CD-ROM
    e. installation via FTP over the Internet

HANDS-ON PROJECTS

Project 8-1

Every network operating system vendor likes to boast about how organizations are using their software to its fullest potential. In this project you will mine the Microsoft Web site to discover how various organizations use networks and their NOS software. You will need a computer with Internet access that has installed on it a modern Web browser program such as Internet Explorer version 4 or higher or Netscape Navigator version 4.5 or higher.

1. If you are not already connected, connect to the Internet and then launch your browser software.
2. Point your browser to the following URL: *www.microsoft.com/windows2000/server/evaluation/casestudies/default.asp*
3. This page lists case studies and success stories of companies that used Windows 2000 Server on their networks. Scroll through the page to get an idea of what type of companies are using this NOS. How many of them look familiar to you? Can you make any generalizations about what industries are most likely to use Microsoft products?
4. Click one of the case studies to read more about it. What services does Windows 2000 Server provide for the company in question (for example, file sharing, printing, communications, network management, Internet services, and so on)? What features does Microsoft seem to emphasize the most? What does that emphasis tell you about their competition with NetWare or UNIX?
5. As you read through the success story, on a separate piece of paper make a note of any terms you do not understand. Which of these terms do you think are Microsoft trademarks and which do you think are general networking terms?
6. Read the entire case study or success story and note the Microsoft products that this customer uses. Besides Windows 2000 Server, what other Microsoft products do they use? Why might it make sense for the customer to use Web server and application programming language software, for example, from the same company that made its NOS?
7. Close your browser window and terminate your Internet connection.

Project 8-2

In this project you will install Windows 2000 Server. You must have a computer that meets the system requirements for Windows 2000 Server, plus the Windows 2000 Server installation CD-ROM and a valid registration key. You will also need the disk that came with the computer's NIC.

1. First, create an installation checklist that identifies the choices you will need to make during installation of this server. (Appendix D provides an example of an installation checklist.) Plan to make your server act as domain controller.

2. Make sure that your server is not networked with any other computers.
3. Install Windows 2000 Server according to the steps provided earlier in this chapter, with the following exceptions:
   - Rather than creating one large partition, create two equal-sized NTFS partitions: one called **SYS** and one called **DATA**.
   - Use the domain name **CLASSX** and call your server **STUDENTY**, where *X* is your pair number (assuming the class can divide into pairs) and *Y* is your seat number.
   - Instead of letting the setup program detect your NIC, when prompted, choose **Select From List** and **Have Disk**. Install the drivers from your NIC configuration disk.
4. When asked for the Administrator password, choose a password you believe to be secure.
5. After the installation is complete, restart the computer and log on as the Administrator.

Project 8-3

In this project you will create a customized MMC console on the Windows 2000 server that you just created.

1. Log onto the Windows 2000 server as Administrator or as a user with equivalent privileges.
2. Click **Start**, and then click **Run**. The Run dialog box appears.
3. In the text box, type **mmc**, then click **OK**.
4. The Console window appears, with a space for the administrative tools listing in the left pane and the details listing for each administrative tool in the right pane. Next you will add snap-in tools that allow you to manage Active Directory.
5. Click **Console** on the menu bar, and then click **Add/Remove Snap-in**. The Add/Remove Snap-in dialog box appears.
6. Click **Add**. The Add Standalone Snap-in dialog box appears.
7. Select the **Active Directory Users and Computers** tool, and then click **Add**.
8. Click **Close** to close the Add Standalone Snap-in dialog box.
9. Click **OK** to save your changes. You return to the MMC console window. Next you will save the console you have created.
10. Click **Console** on the menu bar and then click **Save As**. The Save As dialog box appears. In the File name text box, type a name for your console, such as MyMgr.
11. Click **Save**. Notice that the name of your MMC console window changes to "MyMgr."
12. Close the MyMgr console window.

13. Click **Start** on the menu bar, point to **Programs**, and then point to **Administrative Tools**. Notice that your console appears as a new menu option.

Project 8-4

In this project you will practice creating and configuring a user account on a Windows 2000 server. You should use the server that you installed in Project 8-2 or another properly installed and configured Windows 2000 server.

1. Log onto the Windows 2000 server as Administrator.
2. Follow the steps described earlier in this chapter for creating a domain user account.
3. After creating the account, while in the Active Directory Users and Computers snap-in window, double-click the user account you just created. The user's property dialog box should appear.
4. Note the tabs that appear, and that the **General** tab is selected by default. In the Description text box, enter the job title **Consultant** for this user.
5. Next you will limit the hours that this user can access the network. Select the **Account** tab. Notice the default parameters for this option.
6. Click **Logon Hours**. The Logon Hours dialog box appears, as shown in Figure 8-24.

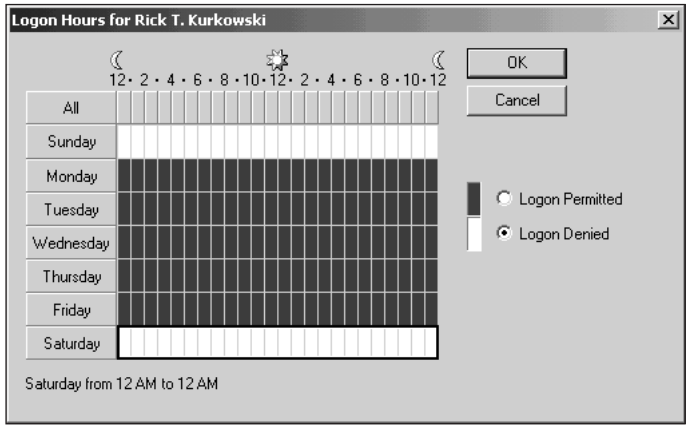

Figure 8-24 Logon Hours dialog box

7. Point to the cell in the last row, first column—the cell that corresponds to 12 A.M. on Saturday. Click and drag the pointer to the far column to select the entire day. Click **Logon Denied**, and then click **OK** to save your changes.
8. Click the **Dial-in** tab of the user's property dialog box. Note the variety of properties that can be associated with a user's dial-in connection to the server.

9. Select the following options in the Dial-in tab: **Allow access**, **No Callback**, and check the **Assign a Static IP address** option. After you select the last option, enter a made-up IP address in the Assign Static IP address text box. What would be the advantage of assigning an employee a static IP address that applies only when she remotely dials in to the server?
10. Click the other tabs in the user's property dialog box to find out what else you can configure.
11. When you have finished, click **Apply** to save the changes you have made to this user account.

Project 8-5

In this project you will have the opportunity to view and modify the virtual memory settings for a Windows 2000 server. This project requires a properly installed and configured Windows 2000 server.

1. Log onto the Windows 2000 server as Administrator or as a user with equivalent privileges.
2. Click **Start**, point to **Settings**, and then click **Control Panel**. The Control Panel window opens.
3. Double-click the **System** icon, click the **Advanced** tab, click **Performance Options**, and then click the **Change** button. The Virtual Memory dialog box appears.
4. If multiple drives are listed in the Drive box, choose the drive that contains your page file. The Paging file size indicates how much hard disk space is available for virtual memory. In the Initial Size text box, enter a number that is double the default number of MB initially available for virtual memory.
5. In the Maximum Size text box, also enter a number that is double the default. What effect do you suppose this change will have on your server's performance? What effect would it have if your server had a low volume of traffic and sufficient RAM to handle its processing needs?
6. Click **OK** to save your change.

CASE PROJECTS

1. A statewide healthcare insurance provider, Evergreen Health, has asked you to help plan a new Windows 2000 network. The organization has been using Windows NT servers until now, but because the organization is now under new management and because Evergreen has just acquired a few smaller health insurance providers, the IT Department wants to rethink the network design from scratch. You meet with a team of employees that includes the IT Director, network administrator, and several network technicians. Unfortunately, none of Evergreen's professionals has had time to learn anything about Windows 2000 Server. Describe what could make Windows 2000 Server particularly well suited to Evergreen's environment.

2. One of Evergreen's network technicians says she doesn't believe the servers are capable of handling Windows 2000 Server. She adds that Evergreen has 12 servers that each contain two 333 MHz Pentium processors, 64 MB RAM, dual NICs, dual power supplies, and 6 GB hard disks. The IT Director, worrying about her budget, indicates that it would be most cost effective to upgrade the existing server hardware, rather than purchase new servers. They ask you to research the total cost of upgrading all of their servers to the minimal Windows 2000 Server requirements. Search the Web for the parts necessary to do this and provide an estimate of how much the upgrades will cost.
3. Even though you were able to provide the IT Director with a cost for the upgrades, you tell the team that you don't recommend leaving all the servers at the minimum hardware requirements. Write down four questions you will ask them about their network environment to help determine which servers might need more system resources.
4. Months later, Evergreen Health has upgraded its hardware and is ready to plan for the installation. The IT department would like your help designing the directory's logical structure. Evergreen contains the departments, locations, and users shown in Table 8-3, below. Users in each department need to share files with users in every other department. Except for the sales offices, users in each location are self-sufficient; that is, they use their location's servers and printers and do not regularly rely on resources in other locations.

Table 8-3 Evergreen Health departments, locations and users

Department	Location	Users
Information Technology	Boston	32
Customer Service	Boise	240 (over three shifts)
Accounting	Boston	20
Claims	New York	38
Operations	Boston	10
Sales	Cincinnati	5
Sales	Boston	10
Sales	Phoenix	8
Sales	San Diego	5
Sales	Chicago	6
Sales	New York	10
Sales	Seattle	7
Marketing	New York	24
Human Resources	Boston	6
Legal	Boston	18

- What type of licensing options should Evergreen purchase?
- Sketch two different domain hierarchies that may work for Evergreen. Include multiple domains and organizational units as you see fit.
- Assign names to the domains and OUs you've created. List the advantages to each of the two models.
- Upon reflection, which domain model would you recommend?
- Insert the following user objects into the San Diego Sales office of your recommended domain model: Sue Anderson, Del Boudreau, Ann Nicastro, Everett Schultz, and Clay Velacruz and assign them user names. Now list the distinguished names for each user, including their OUs and domains.